PHILIPS

PHILIPS data systems

CU SERVICE MANUAL

(Preliminary Version)

for

P850 - P855 - P860

NV PHILIPS - ELECTROLOGICA
COMPUTER SYSTEMS DIVISION
APELDOORN - NETHERLANDS

List of effective pages for:

CU SERVICE MANUAL

(Preliminary Version)

for

P850 - P855 - P860

Front Pages	(i) to (vi)
Section I	1-1 to 1-9
Section II	2-1 to 2-12
Section III	3-1 to 3-11
Section IV	4-1 to 4-8

Date of printing 1st December 1971.

PREFACE

This manual contains the logic and special circuits information needed to service the Device Control Unit (DCU) cards produced by Philips for the P800 series of Mini-Computers. The manual is divided into sections and each section except the first deals with a specific DCU.

The first section deals with the I/O bus and its timing and with the operation of DCU's in general. This section is applicable to all the DCU's included in the following sections.

All the remaining sections deal with specific DCU's and their operation. The last page of each section is a fold-out logic diagram that also includes information on the special circuits. Each diagram has grid reference points marked on it to enable easy location of the various logic elements described in the text. These reference points should not be confused with similar reference points etched on the DCU cards, that are used to identify the logic elements.

The information contained in this book is based on the documentation available at the time of printing 1st December 1971. However, or should any user wish to make any suggestion for improving this manual, he is invited to send this comments to:

MANUAL WRITING SMALL COMPUTERS

Philips C. T. I.

4 à 16 avenue du Général Leclerc

92 - Fontenay-aux-Roses.

France.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

SECTION I

INTRODUCTION

All the Control Unit (CU) cards described in this manual operate in a similar way and the information in this section is applicable to all of them. They are the interface between the CPU and the peripheral device and translate I/O instructions into mechanical actions of the device and provide the sequencing and control signals to effect data transfers.

1.2 I/O BUS Communication between the CPU and the CU is via the I/O bus; its lines being hard wired to the sockets reserved for CU cards. Figure 1.1. shows the connection of CU cards to the bus and up to 32 CU cards can be connected to the I/O bus.

1.3 COMMANDS The commands sent to a CU will be:

(i) CIO - To start or stop a transfer,

(ii) INR - To effect a transfer from a device to the CPU,

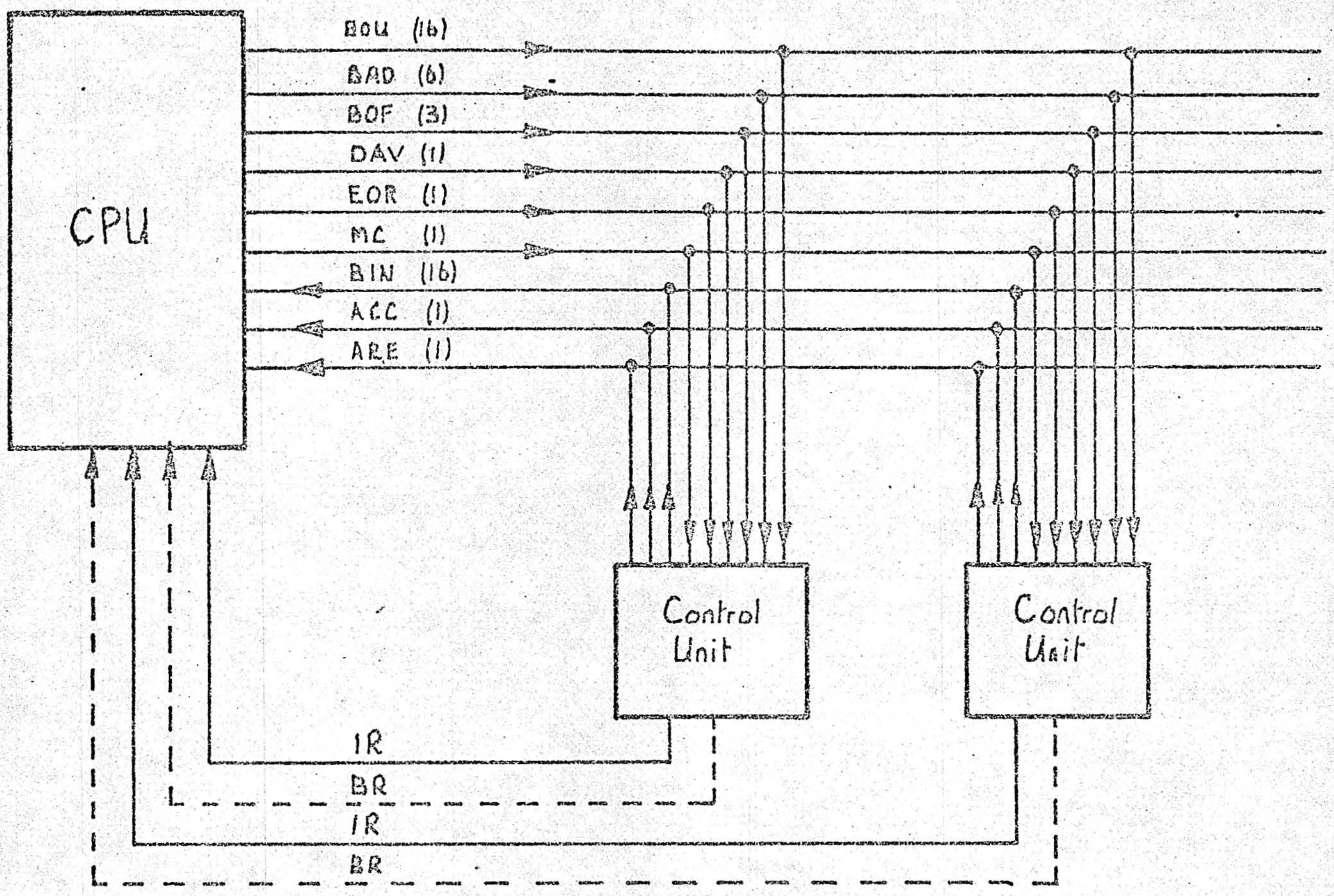
(iii) OTR - To effect a transfer from the CPU to the device,

(iv) TST - To test the status of the device before starting or during any transfer,

(v) SST - To test the status of the device after the transfer has been completed.

These commands originate in the I/O instructions and will include the address of the device.

1.4 ADDRESSING. The CU address is sent via the BAD lines of the I/O bus. Each CU card will have a BAD input gate that will only recognize its own address. The code for this address will have been wired up on the pinboard of each CU card. Some CU cards can control more than one device, and these cards have the address of each device wired up on their pinboard.



Key:

CPU Output:
- BOU ... Output data bus (16 lines)
- BAD ... Device Address bus (6 lines)
- BOF ... Operation Function bus (3 Lines)
- DAV ... Device Address Valid bus
- EOR ... End of Range bus *

CU Output:
- BIN ... Input data bus (16 lines)
- ACC ... Accept Command line
- ARE ... Address Recognised line
- IR ... Interrupt Request line
- BR ... Break Request line *

* Only used with Multiplex transfers

Figure 1.1. CU to I/O Bus Connection

1.5 FUNCTION CODE This code contains the actual command to the CU and is sent via the BOF lines of the I/O bus. The three bits of the code are translated into a command, which is sent to the device via the sequence control part of the CU. The following table shows the decoded bits translated into commands:

Instruction	Bit 4	Bit 8	Bit 9	Command
CIO	0	1	1	Start
CIO	0	1	0	Stop
INR	1	0	0 or 1	Input Transfer
OTR	0	0	0 or 1	Output Transfer
TST	1	1	n/s 0 ?	Test Status
SST	1	1	1	Sense Status

These commands will only be accepted if the control signals have validated the address and the CU sequence logic is in the right mode.

1.6 CONTROL SIGNALS There are three control signals lines from the CPU to the CU and two control signal lines from the CU to the CPU contained in the I/O bus. In addition there will be the Interrupt and/or the Break Request lines from the CU to the CPU.

DAV This signal is used to validate data on the BAD and BOF lines. If the address on the BAD lines is recognized, the CU will respond with the ARE signal and if the command on the BOF lines is accepted it will respond with the ACC signal.

MC This is the Master Clear signal that is activated either from the CPU control panel or the power supply each time the CPU is switched on.

EOR This is the End of Range signal which is sent at the end of a Multiplex transfer.

ARE This signal is sent to the CPU when the address on the BAD lines has been accepted by the CU.

ACC This signal is sent to the CPU when the command on the BOF lines has been accepted.

INTERRUPT and BREAK REQUEST These signal lines connect each discrete CU to the CPU interrupt system. The Interrupt line is used for programmed channel transfers, the Break Request line for multiplex transfers. Each is activated by the CU to indicate to the CPU that it is ready to exchange data or status information.

1.7 I/O BUS TIMING The timing of the I/O bus signals are shown in figure 1.2.

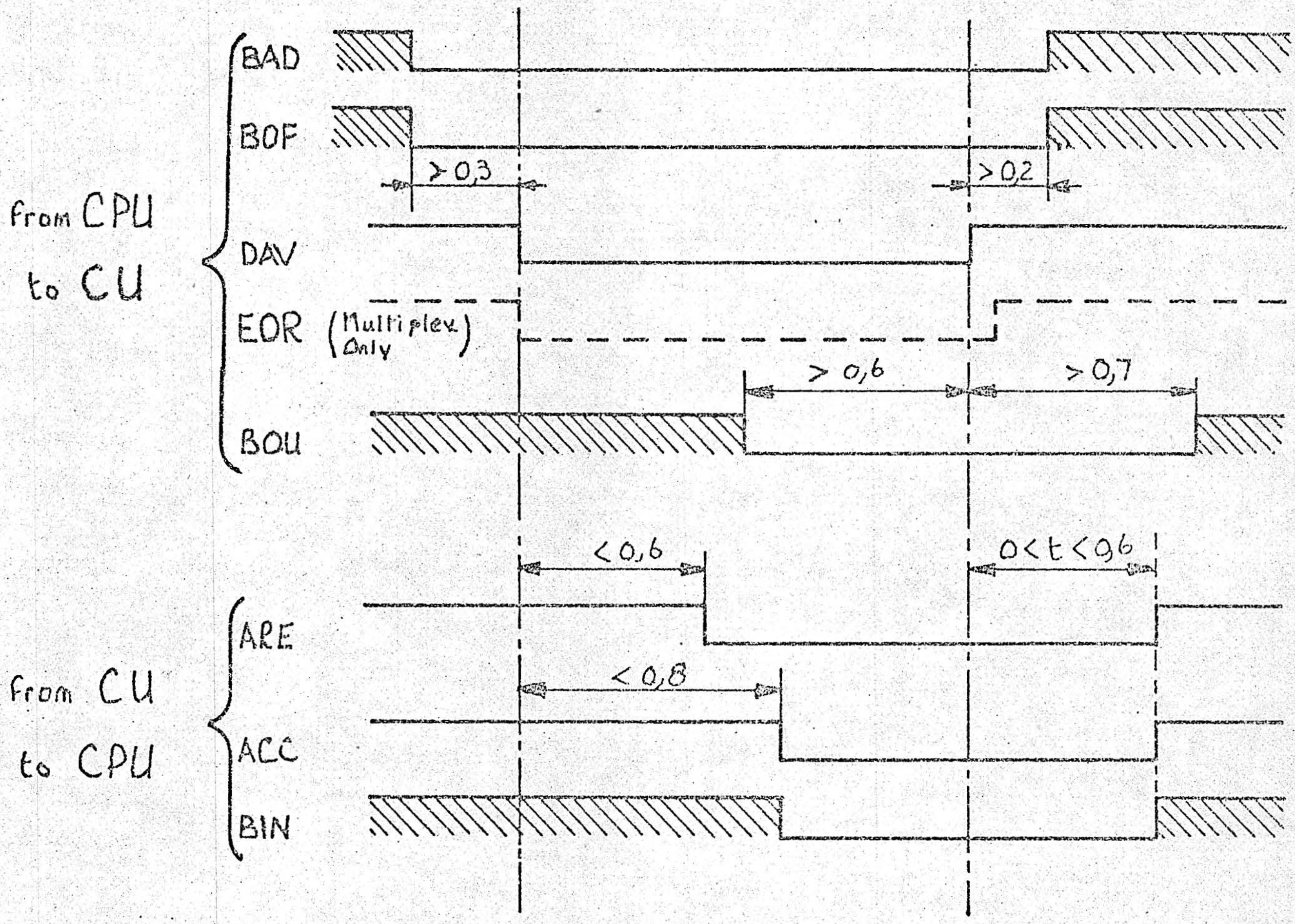
1.8 CU MODES Figure 1.3 shows a block diagram of a typical CU. Once the address has been accepted by the CU, a check is made to find out if the CU is in the right mode to accept the command on the BOF lines. The modes are:

INACTIVE The only commands accepted in this mode are CIO start and TST. In this mode a TST command will cause the CU to respond with zeros on all the BIN lines. A CIO Start command will cause the CU to switch from inactive to the Execute or Exchange modes depending upon whether the exchange is to be input or output.

EXECUTE This mode causes the device to perform a single mechanical action after which it switches the CU into either the Exchange or Wait Status modes. The only commands accepted in this mode are CIO Halt and TST. The CIO Halt will cause the CU to switch to the Wait Status mode. The TST will not affect the CU operation but will send a busy signal signal back to the CPU.

EXCHANGE This mode activates the Interrupt or Break Request line to indicate to the CPU that it is ready to receive or send data. After the exchange the CU will switch back to the execute mode. The only commands accepted in this mode are INR, OTR or TST. The INR will transfer data to the CPU via the BIN lines, the OTR will accept data from the BOU lines and the TST will send the CU status back to the CPU.

WAIT The CU goes into this mode after receiving a CIO Halt command whilst in the Execute mode. When the SST command has been accepted and serviced, it will switch to the Inactive mode. The only commands accepted in this mode are TST and SST. In both cases the CU sends the device status back to the CPU via the BIN lines;.

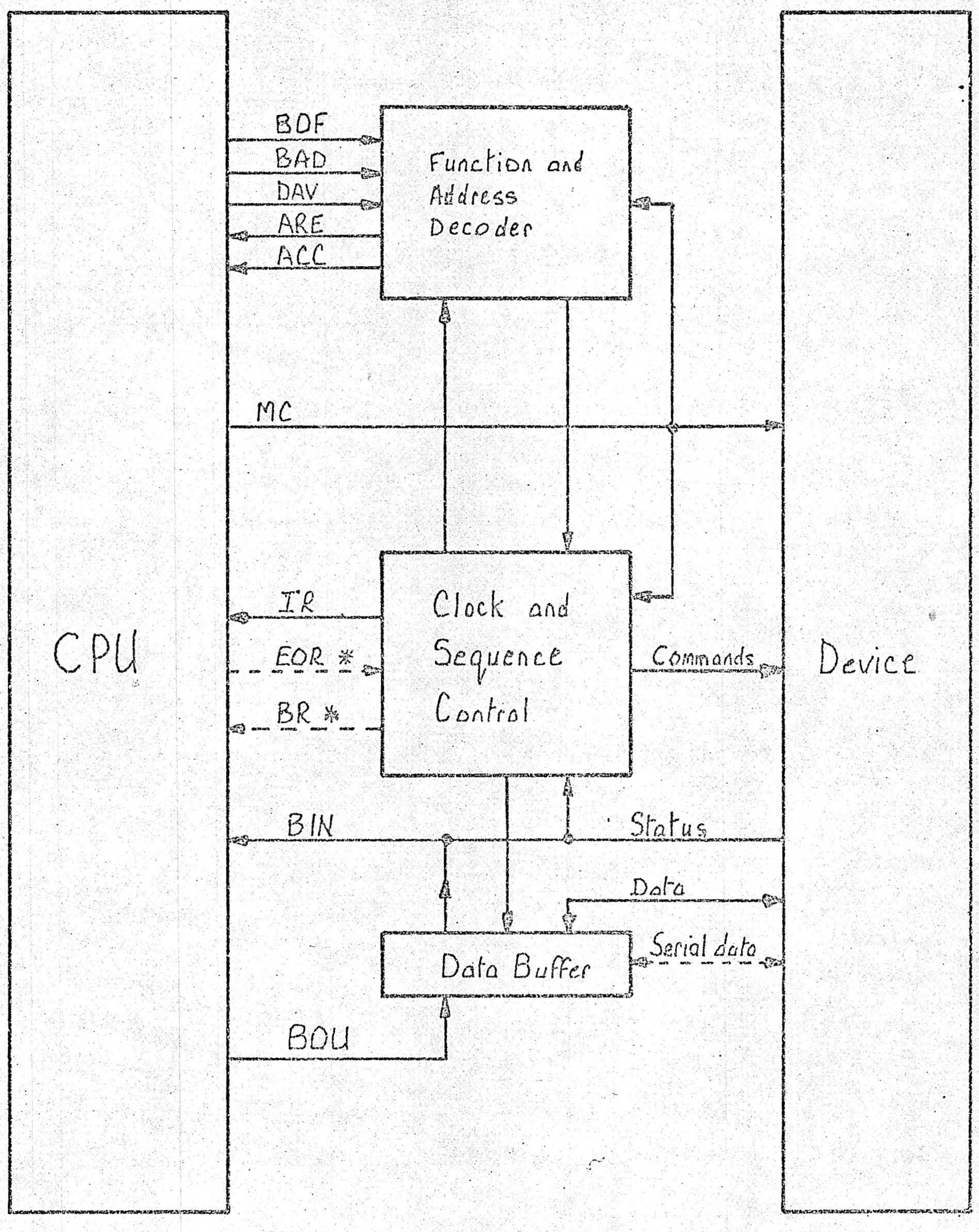


: Not defined

Times are given in microseconds

Low levels are active or data 1 except BOU where high level is data 1

Figure 1.2 I/O Bus Timing



*: Only used by multiplex

Figure 1.3 Block Diagram of Typical C.U

1.9 MODES SEQUENCES The mode sequence depends on whether the transfer is input or output. Both sequences are shown in the flowchart in Figure 1.4.

1.10 CU AND DEVICE STATUS The status of the CU is tested by the TST command and the device status by the SST command.

TST This command is accepted by the CU in any mode. If the CU is in the Inactive mode (not busy) it will respond by sending all zeros to the CPU via the BIN lines. In any other mode (busy) it will respond by sending a 1 bit on BIN line 15 other lines are not significant.

SST This command is only accepted in the Wait mode and responds by sending the status of the device to the CPU via the BIN lines. The most common status bits are:

bit 15: Manual intervention required by the operator
bit 14: Throughput Error
bit 13: Data fault
bit 12: Incorrect length

Other bits may be used by the more sophisticated control units, but these will be given in the section dealing with these units.

1.11 I/O BUS CONNECTIONS
Connections to the pins of the I/O bus sockets have been standardised so that any DCU card can be plugged into any socket without modification. The list of connections and signal names is given below:

Socket		Socket	
1A01	+5V	2A01	BOF02/
1A02	reserved	2A02	EOR
1A03	BIN00/	2A03	BOF01/
1A04	BIN01/	2A04	BOF00/
1A05	BIN02/	2A05	BOU14
1A06	BIN03/	2A06	BOU13
1A07	BIN04/	2A07	BOU12
1A08	BIN05/	2A08	BOU09
1A09	BIN06/	2A09	BOU10

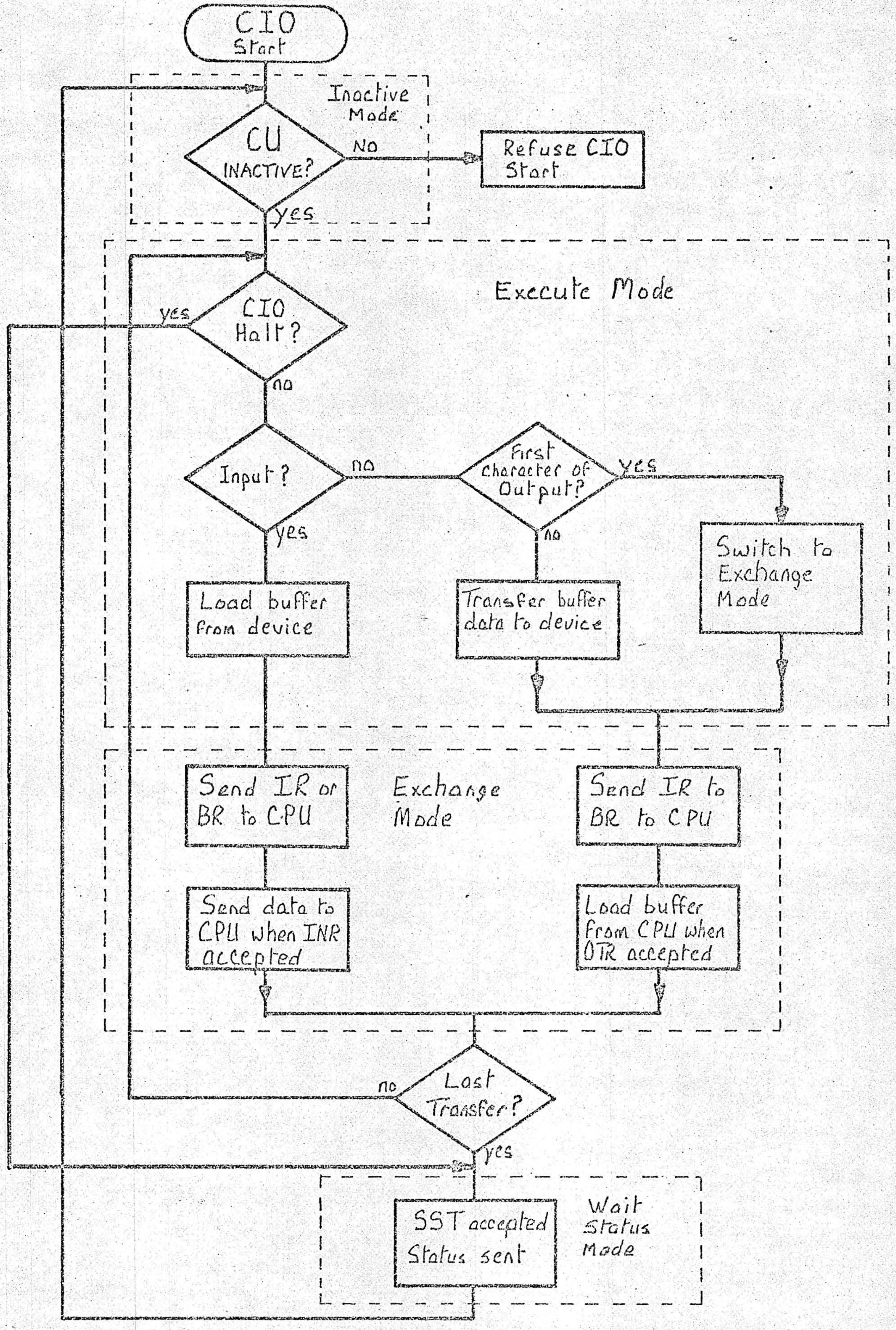


Figure 1.4 Mode Sequences

1A10	BIN07/	2A10	BOU11
1A11	ground	2A11	BOU15
1A12	reserved	2A12	MC/
1A13	ACC/	2A13	BAD03/
1A14	ARE/	2A14	DAV/
1A15	BRIR/	2A15	BAD04/
1A16	BR1/	2A16	BAD05/
1A17	BR2/	2A17	BOU08
1A18	BR3/	2A18	BOU07
1A19	BR4/	2A19	BOU06
1A20	reserved	2A20	BOU05
1A21	BIN08/	2A21	BOU04
1A22	BIN14/	2A22	BAD00/
1A23	+ 6 v	2A23	BOU03
1A24	- 6 v	2A24	BAD01/
1A25	- 12 v	2A25	BAD02/
1A26	BIN13/	2A26	BOU02
1A27	BIN15/	2A27	BOU01
1A28	BIN09/	2A28	BOU00
1A29	BIN10/	2A29	+ 24 v
1A30	BIN11/	2A30	- 5 v
1A31	BIN12/	2A31	+ 5 v
1B31	ground	2B31	ground

SECTION II

ASR CONTROL UNIT

BRIEF DESCRIPTION

The Control Unit (CU) logic and special circuits are contained on one card and will control one ASR. Its main function is to interface the parallel operation of the CPU I/O bus and the serial operation of the ASR. A clock circuit on the card provides the timing pulses necessary to effect transfers of data and to control the sequence of operation.

Standard I/O instructions are used to initiate data transfers and to send control signals to the ASR via the CU. The control signals are the start/stop signals to the Tape Reader and Tape Punch attached to the ASR. These signals are not generated by the CU but are programmed and transferred like any other data.

Figure 2. 1 shows a block diagram of the main logic and special circuits of the ASR CU; the logic drawing for the CU card will be found in the pocket at the back of this book, a smaller logic drawing will be found at the end of this section. The smaller drawing has grid reference marks which are used in the text to enable the logic elements to be found quickly. The following paragraphs of this section describe the operation of the control unit in more detail, with the aid of these two diagrams.

2. 1 ADDRESSING

The address of the CU card can be wired to recognize any address using the AD0 to AD5 (ref: A and B/1, a-3) switch connectors. When the CPU puts any CU address on the BAD lines, it validates it by sending the DAV (ref: A/1) signal. If the address corresponds to the code set up on the switch connectors, the CU responds by sending the ARE (ref: D/1-2) back to the CPU; this signal is also used to enable logic on the CU card.

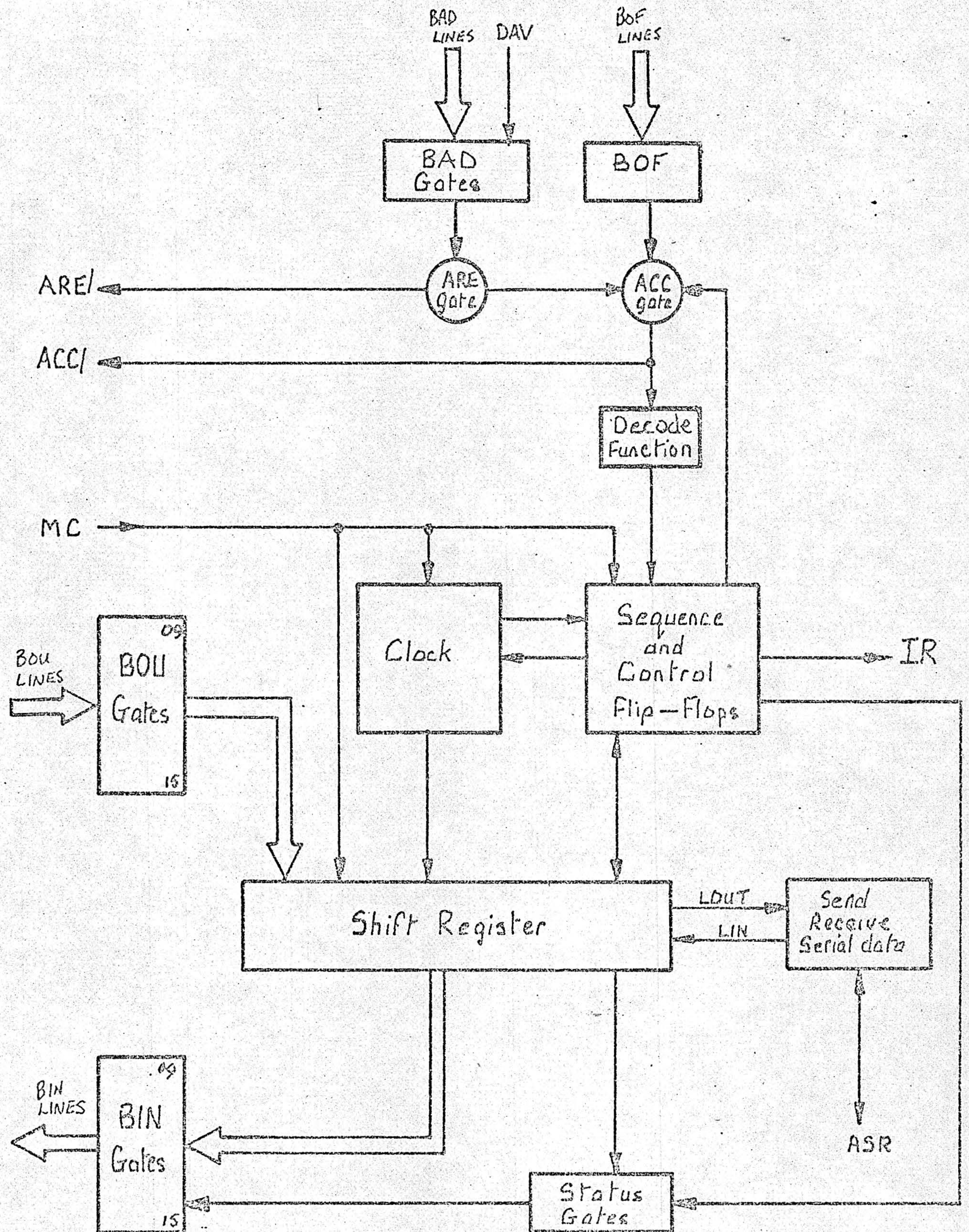


Figure 2.1 Block Diagram of ASR CU

2.2 BOF

Once the address has been recognized, the code on the BOF (ref: A and B/4-5) lines and the present state of the CU are checked to find out if the command can be accepted. If accepted the CU will send the ACC (ref: D/1) signal back to the CPU. At the same time the output from the function decode logic will enable the appropriate logic circuits in the CU to start execution of the command.

2.3 FUNCTION REGISTER

This register is made up of four flip-flops that store the type of function sent on the BOF lines. These flip-flops are:

FCA This flip-flop (ref: E and F/4) remembers that a CIO Start command has been accepted and performed. It enables the sequencer flip-flops to switch to either the Execute (EXT) state if the CIO was an input command, or to the Exchange (ECH) state if the CIO was an output command.

Set During the acceptance of the CIO start command.

Reset During acceptance of a CIO stop command.

(ii) At the beginning of any serialization.

(iii) During the exchange of data with the CPU.

FAOV This flip-flop (ref: F/4-5) remembers that either an INR or OTR command has been accepted and is used to enable the sequencer logic.

Set During any data exchange with the CPU.

(ii) During acceptance of an SST command.

Reset During execution of a Stop command.

(ii) During the Start of any I/O command execution

(iii) At the beginning of any serialization.

FHALT This flip-flop (ref: E/3) remembers that a stop command has been performed and enables the sequencer to switch to the Wait Status (WST) state if FSER is reset. It also inhibits any exchange request; remembers a detected throughput error and in conjunction with FAOV enables the sequencer to switch to the inactive (INCT) state after execution of an SST command.

Set During acceptance of a stop command.

(ii) During the serialization sequence (input) if a throughput error has been detected.

(iii) During the exchange state if the ASR is used on the multiplex chan channel.

Reset During the acceptance of an SST command.

FOUT This flip-flop (ref: B and C/8) controls the operating mode of the serializer circuit. When set it allows transfers from the CPU to the ASR, and when reset it allows transfers from the ASR to the CPU.

Set During the acceptance of a CIO Start command in output mode.

Reset During the acceptance of a CIO Start command in input mode.

2.4. CONTROL FLIP-FLOPS

These three flip-flops control the transfer of data between the CPU and the ASR. They are:

FSER This flip-flop (ref: E and F/6) enables serialization by starting the CU clock counting.

Set During input mode by the Start pulse from the ASR and during output mode by FFF/

Reset At the end of serialization by the trailing edge of the eleventh BP pulse

PE This flip-flop (ref: F and G/6) enables the shift register to either parallel load or one bit shift.

Set It is set by FSER

Reset By the trailing edge of the first AP pulse.

FTHR This flip-flop (ref: E/7) can detect a throughput error when the CU is in input mode.

Set When a new character tries to load the shift register whilst the CU is in the ECH state.

Reset During the acceptance of an I/O command.

2.5 INTERFACE SEQUENCER

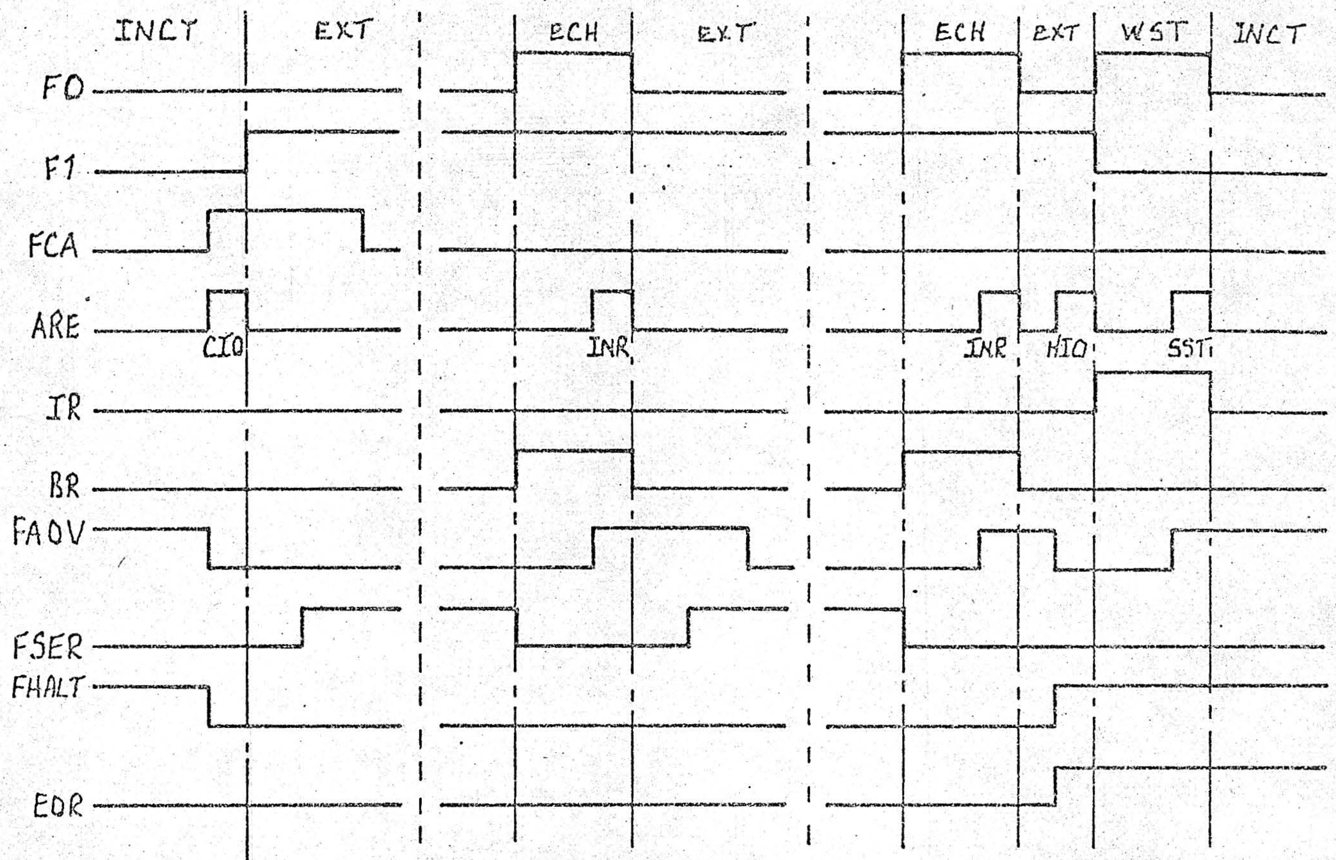
The sequencer uses two flip-flops F0 (ref: E and F/1-2) and F1 (ref: F/3) to switch the appropriate sequence states of the CU during data transfers. A combination of outputs from these flip-flops drives logic gates that provide the signal levels which indicate the state of the CU. These are:

(i) Inactive State (INCT) - F0/. F1/
(ii) Execute State (EXT) - F0/. F1
(iii) Exchange State (ECH) - F0. F1
(iv) Wait Status State (WST) - F0. F1/

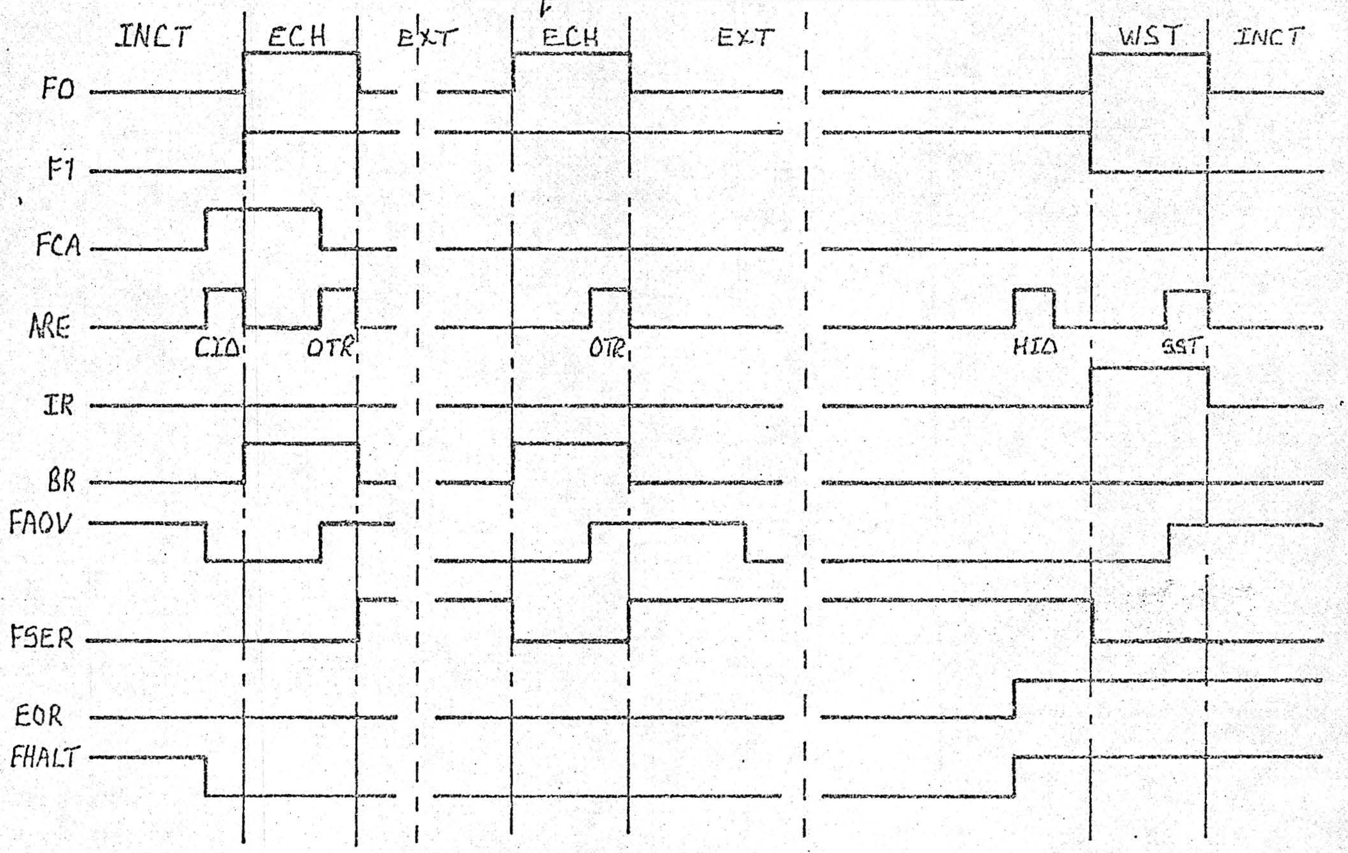
2.6 SEQUENCER AND CONTROL TIMING

The timing for the sequencer and control flip-flops is dependant upon two other sources of timing signals; the I/O bus signals which are controlled by the CPU and the CU clock which controls transfers between the CU and the ASR. It is therefore not possible to draw a timing diagram that will relate the signals to the actual time taken for a transfer. However, figure 2.2. does show the relationship between the signals, in both input and output modes, during a data transfer.

Sequence for INK Command



Sequence for OTR Command



Notes: 1. The section between the continuous dotted lines will be repeated as necessary until the end of data transfer.

2. EOR will only be used when the ASR is used on the Multiplex channel.

3. The IR and BR signals are "ored" together when the ASR is used on the I/O bus.

Figure 2.2 Sequence and Control Timing

2.7 CU CLOCK CIRCUIT AND TIMING

The clock provides timing pulses to synchronize the operation of the shift register and control logic of the CU to the speed of ASR distributor. The basic clock comprises the two monostable modules A3 and B4 (ref: B/6) which are interconnected to form a stable oscilator. Module A4 (ref: B/7) is used to give the correct pulse width to the two outputs. These outputs are used in conjunction with gates B4 (ref: B and C/7) to produce the biphase clock pulses AP and BP.

STARTING THE CLOCK The clock is started when signal FSER is rising after the flip-flop has been set.

STOPPING THE CLOCK The trailing edge of the eleventh BP pulse is used to reset FSER. The effect of FSER going low is taken through a delay network of gates C3 and B4 (ref: C/5-6) and used to stop the clock.

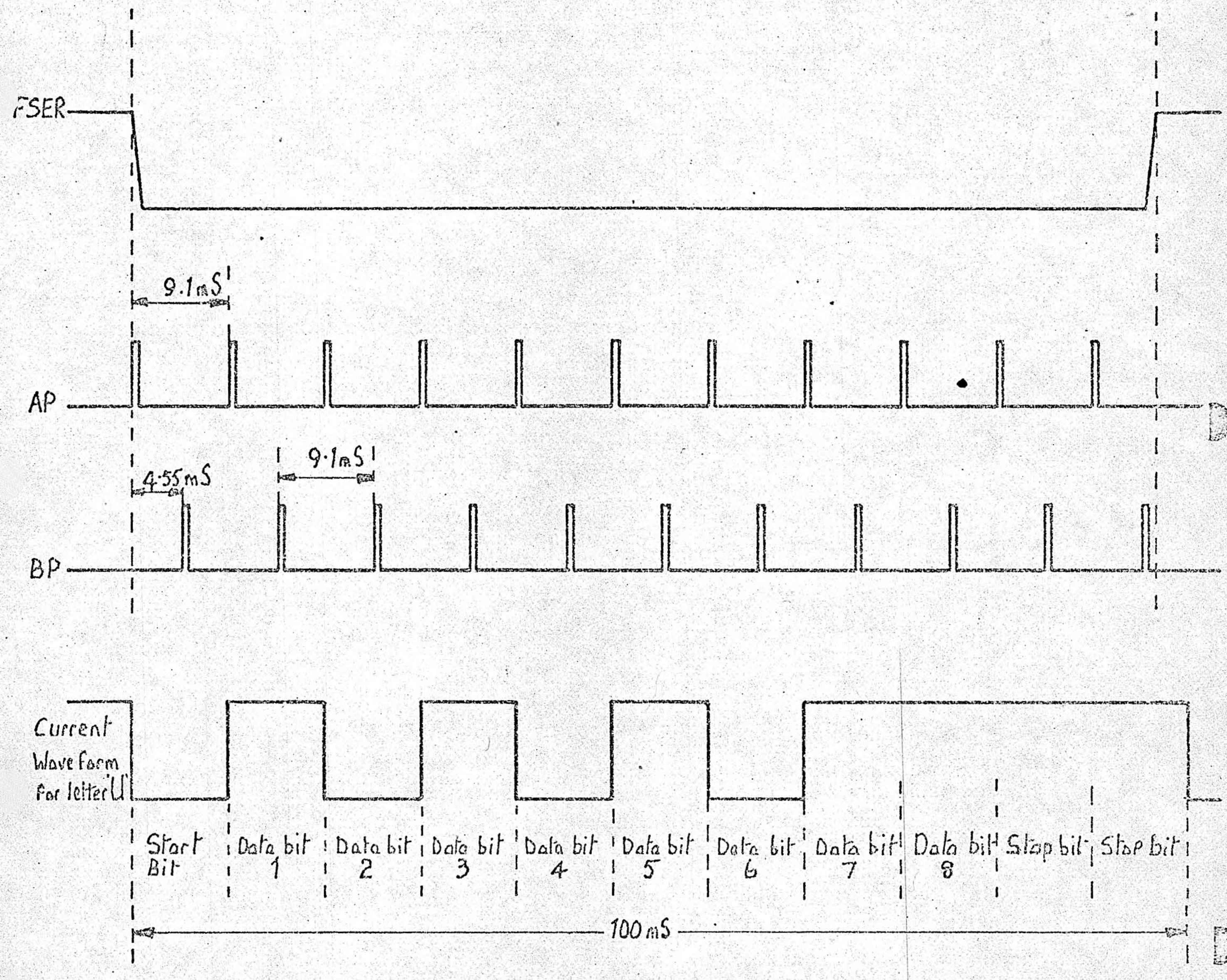
TIMING Figure 2.3 shows the timing of the AP and BP pulses in relation to the transfer of the character U.

2.8 SHIFT REGISTER

The register comprises three 4-bit modules F2, G2 and H2 (ref: B/10, C/9-10 and C/9) that can operate in either paralllel or one-bit shift modes. Only eleven bits of the register are used by the CU and figure 2.4 shows the inter-relationship of the three modules.

OPERATION The operation mode is determined by PE/. When this signal is present the register operates in the parallel mode. This mode is for all transfers between the register and the CPU. When PE/ is not present , the register operates in the one-bit shift mode the timing pulses from the CU clock providing synchronization with the ASR.

INR Command The register shifts one position during each BP pulse of the loading sequence (11 pulses in all). Input to the register is via B0 and the signal level LIN. Loading is initiated by the start pulse from the ASR which will be a low; then the eight character bits are



Notes: 1. The AP and BP pulses are both 400 nano-seconds wide.

2. Data bit 8 will always be high for any character.

Figure 2.3 CU Clock Timing

loaded followed by two stop pulses which will both be high. End of serialization is detected when the start pulse appears at Q10. This level is used via gates C4, G6 and D6 (ref: D and E/6-7) to reset FSER.

OTR Command The character on the BOU lines is loaded into register positions B3 to B9. The most significant bit of the character will always be a 1 so it is not transferred from the BOU lines but is inserted into register position B2 by the register itself. Position B10 is reset to represent the start bit and the other two positions (B0 and B1) are set to represent the two stop pulses. Output from the register to the serializer is from Q11 and as each pulse is sent to the ASR B0 is reset. End of serialization is detected when Q1 to Q10 have all been reset. These levels are used to produce level EMPTI1 which is used to reset FSER.

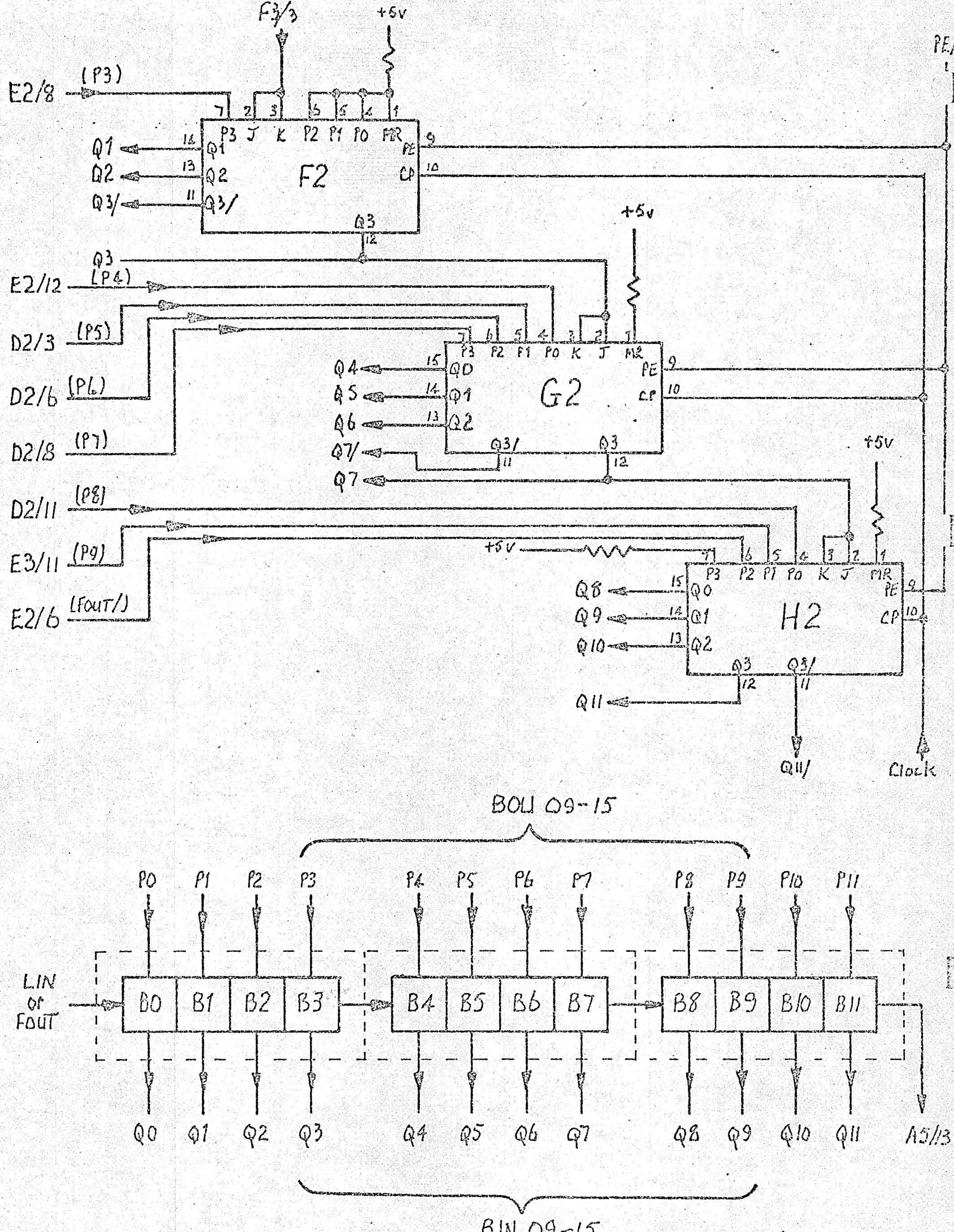
2.9 SERIALIZER

This circuit comprises T1, T2 and T3 (ref: D to G/7-8) and it is connected to the ASR via a single, twisted twin, cable. Data and control characters are sent along it as mark/space current wave form. A current of 20 milliamps represents a mark and a current of less than 3 milliamps represents a space. The relationship between the character U and the CU clock can be seen in figure 2.3.

2.10 STATUS RESPONSES

There are two types of status response one for the TST command and one for the SST command, and both use the same logic to send the response to the CPU. This logic is the G3 gates (ref: B and E/9) that load the BIN lines with the status. The same logic is also used during data transfers to the CPU to load BIN lines 14 and 15 with data.

TST The two responses to this command are inactive or busy. If the CU is inactive the response will be all lows on the BIN lines, if busy the CU will send a high on BIN 15.



Notes:

1. Q3 to Q9 also go to other gates as well as to the BIN lines.
2. Q8 and Q9 go to BIN 14 and 15 via the PARTBIN logic gates.

Figure 2.4 Shift Register

SST The response after a successful transfer will be all lows on the BIN lines. If a throughput error occurs during a transfer, the CU will send a high on BIN 14.

All other bits are insignificant for both types of command.

2.11 EXCHANGE CONTROL SIGNALS

These are the PIL/, BRL/ and EOR signals. The first two are used to ask the CPU to effect a data exchange, the last one tells the CU that the last character has been sent.

PIL/ This signal is active during the WST state and asks the CPU to send an SST command.

BRL/ This signal is active during the ECH state and asks the CPU to send either an INR or an OTR command. This signal is normally used only during multiplex exchanges, but when the ASR is used on the programmed channel the BRL/ and PIL/ signals are 'ored' together on the I/O bus.

EOR This signal is only used during multiplex exchanges to indicate to the CU that the last character has been either sent or accepted.

2.12 ASR TO CU CONNECTION

Both input and output data are transferred, between the CU and the ASR, via a pair of twisted wires. One of the wires is connected to pin A02 (logic level) and the other wire to pins A05 and A06 (ground) of an ELCO connector socket. This socket is plugged onto the CU card and the other ends of the twisted pair are connected to the data input terminals of the ASR.

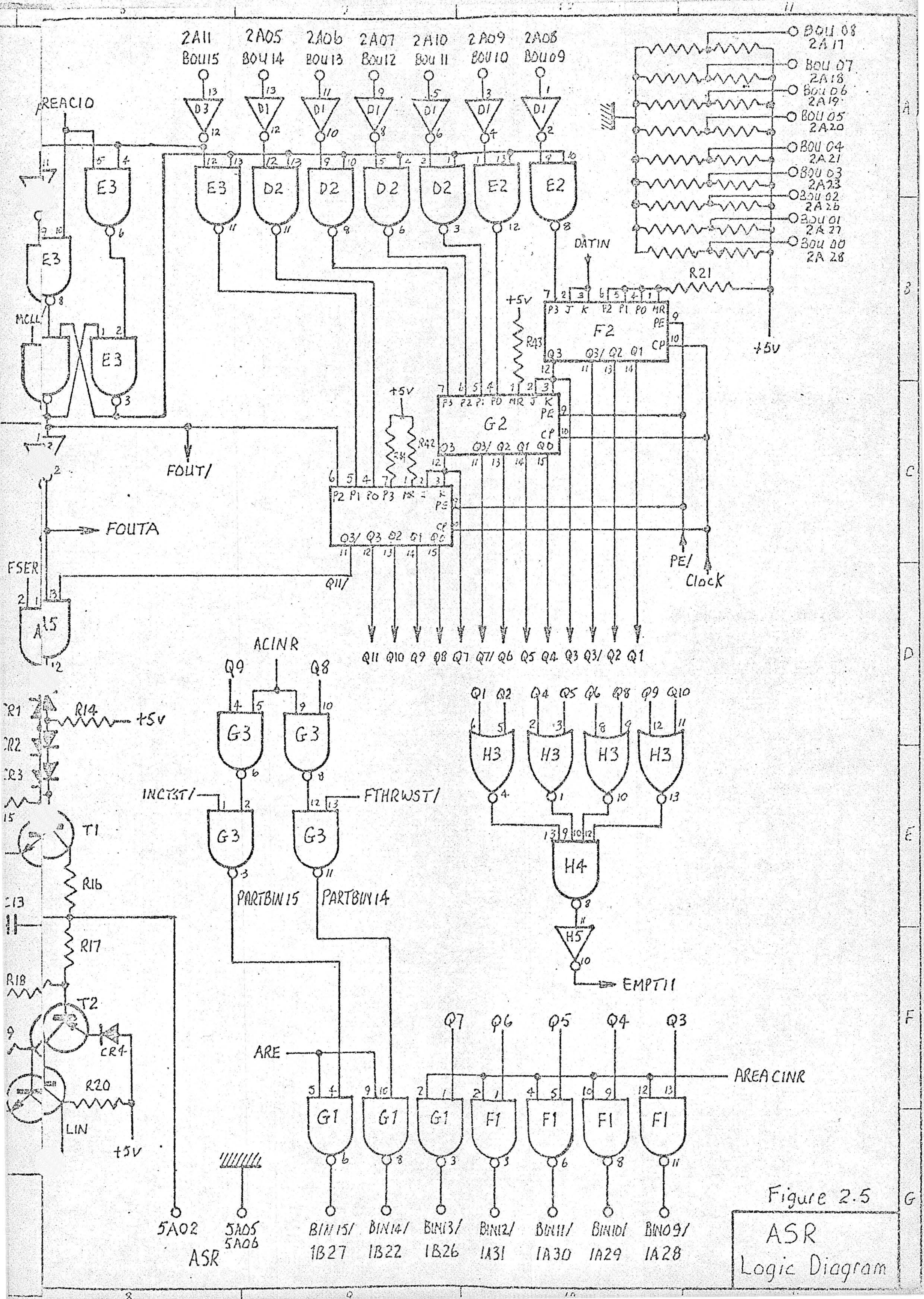
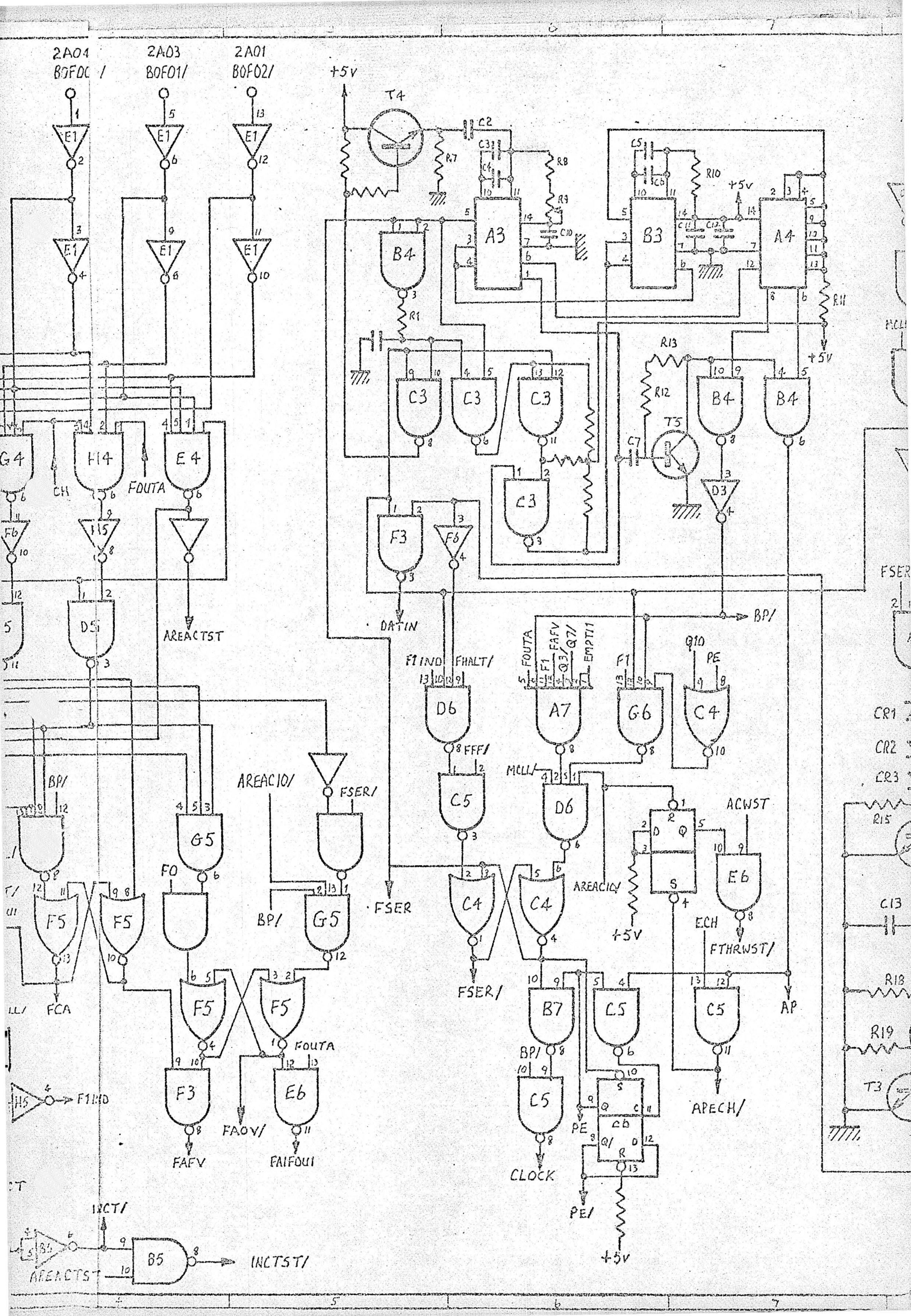
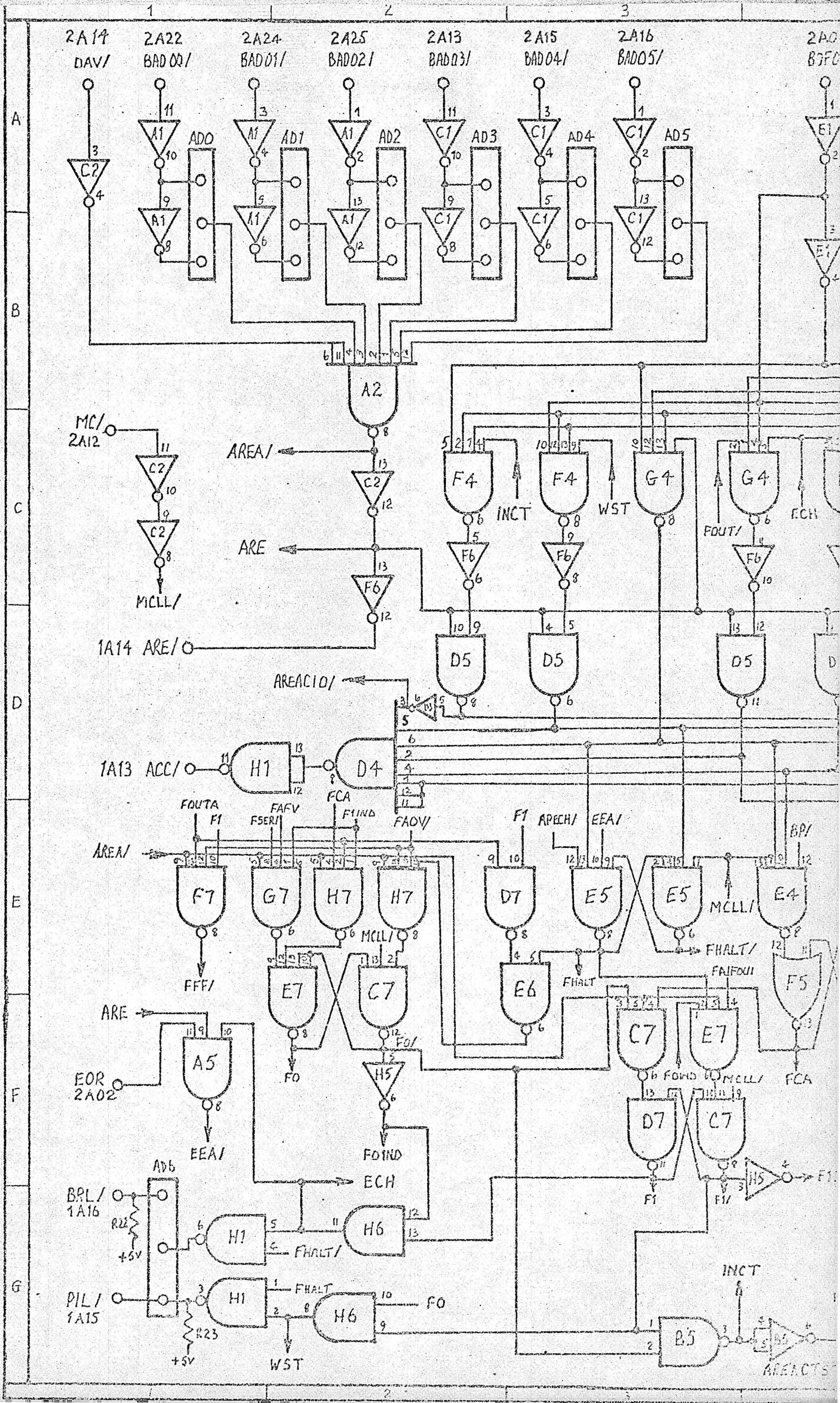


Figure 2.5

ASR Logic Diagram


2A04
BOFDC /
2A03
BOFO1/
2A01
BOFO2/
+5V
T4
C2
C3
C4
R7
R8
R9
C10
A3
B3
C5
C6
R10
A4
R11
R1
B4
C11
C12
R13
R12
T5
C7
D3
BP/
DATIN
F3
F6
E1
G4
H14
E4
CH
FOUTA
H5
D5
AREACTST
G5
F5
FO
FCA
FAFV
FAOV/
E6
FAIFOUI
F1IND
INCT/
B5
INCTST/
F1IND
FHALT/
D6
FFF/
F1
Q3/
Q7/
EMPTI1
A7
G6
Q10
PE
MCLL/
AREACIO/
FSER/
FSER
ACWST
FTHRWST/
ECH
AP
APECH/
B7
CLOCK
PE/
CR1
CR2
CR3
R15
C13
R18
R19
T3



1
2
3
A
B
C
D
E
F
G
2A14 DAV/
2A22 BAD00/
2A24 BAD01/
2A25 BAD02/
2A13 BAD03/
2A15 BAD04/
2A16 BAD05/
AD0
AD1
AD2
AD3
AD4
AD5
A1
C1
C2
E1
A2
AREA/
ARE
MC/ 2A12
MCLL/
1A14 ARE/
F4
G4
F6
D5
INCT
WST
FOUT/
ECH
AREACID/
1A13 ACC/
H1
D4
FCA
FOUTA
F1
FAFV
F5ER/
F1IND
FAOV/
APECH/
EEA/
BP/
F7
G7
H7
D7
E5
E4
FFF/
MCLL/
E7
C7
E6
FHALT
FHALT/
F5
FO
FO/
H5
FOIND
FCA
A5
EOR 2A02
EEA/
AD6
ECH
BRL/ 1A16
R22
+5V
H6
PIL/ 1A15
R23
WST
FO
B5
AREACTS

SECTION III

PUNCHED TAPE/CARD READER CONTROL UNIT

BRIEF DESCRIPTION

This control unit can be used to drive either a punched tape reader or a card reader. All the logic and special circuits are contained on one card and only one device can be connected to a card.

Figure 3.1 shows all the main logic blocks and data paths of the CU. The CU is connected to the CPU via the standard I/O bus. All I/O bus signals and their timing rules are detailed in Section I of this book. Connection between the CU and the device is via special circuits CSO and CSI and all these signal levels are active high.

When the CU is used with a tape reader, CSO will send the FWD (tape forward) and STOP (tape stop) commands to the device and CSI will interface the eight data channels, the strobe pulse and device status signals with the CU logic circuits.

When the CU is used with a card reader, the CSO signal FWD is used as the POC (pick one card) signal to the device; the stop signal is not used. The CSI circuits interface the twelve data channels, the strobe pulse and the device status signals to the CU logic circuits.

Operation of the CU is programmed by standard I/O instructions and data exchanges can be via either the programmed or multiplex channels depending upon the type of CPU.

At the end of this section is a fold-out drawing of the logic elements and special circuits that make up the CU. The grid reference points of this drawing are used in the following paragraphs to enable easy location of the relevant logic elements or special circuits. The CU functions in a similar way for both the tape and card reader so to avoid repetition, the following paragraphs apply to both devices unless otherwise stated.

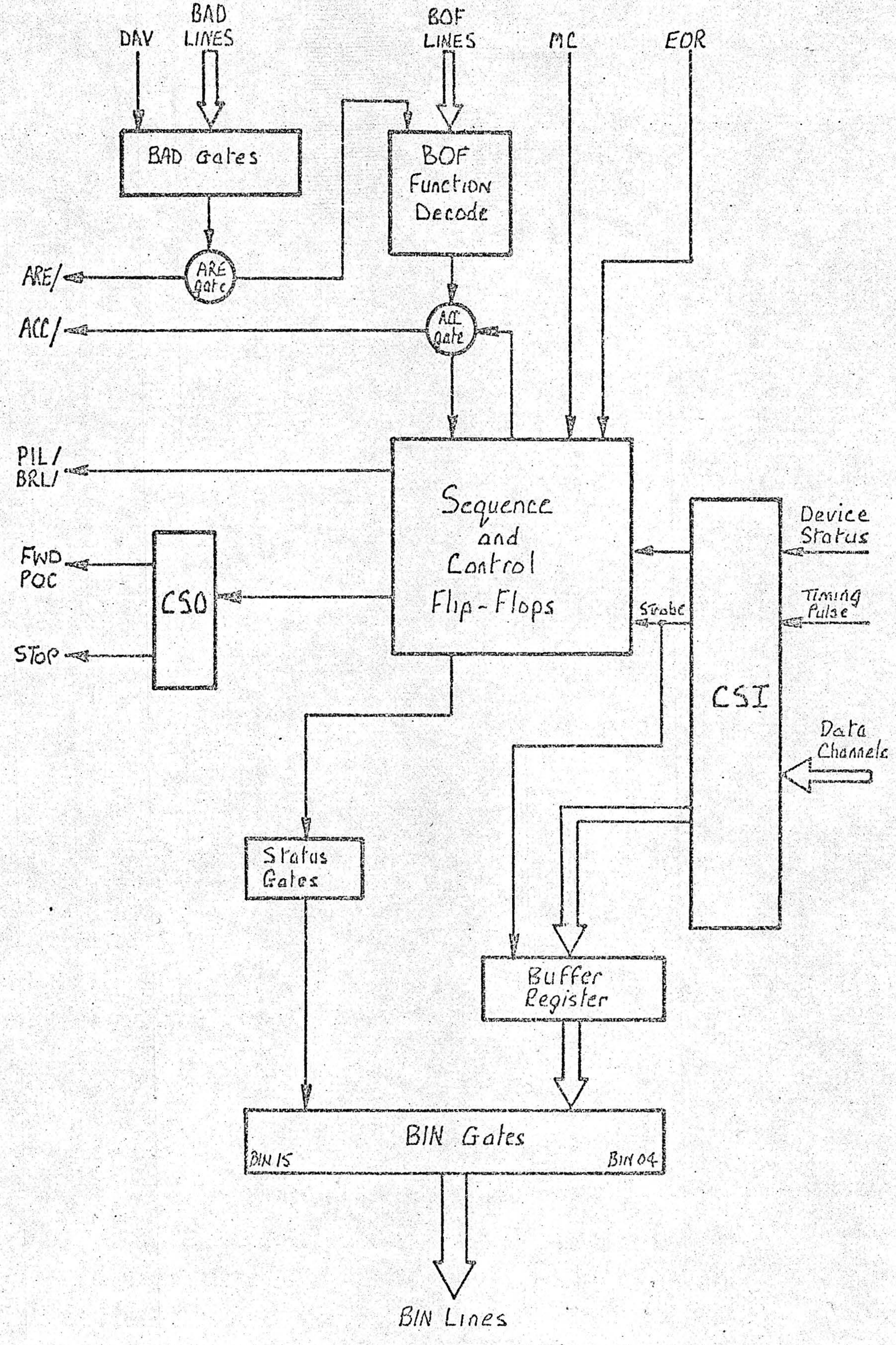


Figure 3.1 Block Diagram of Tape / Card Reader C.U.

3.1 ADDRESSING (BAD lines)

The CU can be wired to recognize any desired address using switch connectors AD00 to AD05 (ref: B/1-4). The outputs from the connectors together with the DAV/ signal, from the CPU, activate gate A2 (ref: C/2-3) producing AREA/ when the addr-ss is recognized. This signal is used to enable the BOF decoder, the interface sequencer and via the inverter B2 and gate G2 the ARE/ signal that is sent back to the CPU.

3.2 FUNCTION DECODING (BOF lines)

Data from the BOF lines is decoded by a 9301 module D1 (ref: B/4-5). The inputs to the module are validated by AREA/ and the outputs obey the following truth table:

INPUT	A2	A1	A0		OUTPUT
CIO	0	1	1	4/	START/
HIO	0	1	0	5/	ACHALT/
INR	1	0	*	2/3/	INRA/INRB/
SST	1	1	1	0/	SST
TST	1	1	0	1/	ACTST/

* can be either O or 1 depending on the function.

The outputs from the module are used to activate the control and sequence logic and in addition produces the ACC/ signal, via gates D2 and G1, that is sent back to the CPU.

3.3 INTERFACE SEQUENCER

This logic controls data transfers by switching the CU into the appropriate state. It is achieved by using a combination of the outputs from two flip-flops, F0 element A4 (ref: E/3) and F1 element A4 (ref: E/4).

INCT: is the inactive state and both F0 and F1 are reset. It is produced by either an SST command or the MCL/ signal. In this state the CU is waiting for a CIO start command to switch it into the execute state.

EXT: is the execute state and allows one character from the tape reader, or one column from the card reader to be read into the buffer register. It is entered when F0 is in the reset condition, and F1 has been set by either ACIO or ACINR. If the F0 flip-flop is set by gate C4 (ref: D-E/3), the CU will switch to exchange state; but if the F1 flip-flop is reset by gate C4 (ref: D-E/4) the CU will switch to the wait status state.

ECH: is the exchange state and is produced when both F0 and F1 are set. It allows the CU to ask the CPU for a data exchange by activating BRL/ via gate E1 (ref: E-F/2-3). The CPU will respond by sending an INR command which will activates gates C3 (ref: C/5) producing ACINR/. This signal is used to gate data on to the BIN lines and, via gate B3 (ref: E/5), to reset flip-flop F0 so that the CU can switch back to the EXT state to allow another character or column to be read from the device.

WST: is the wait status and is produced when F0 is in the set condition and F1 has been reset via gates D4 and C4 (ref: D/4-5). It allows the CU to send an interrupt to the CPU which responds with an SST command. This command causes the CU to send the status word back to the CPU, via the BIN lines, and resets flip-flop F0 so that the CU can switch back to the INCT state.

3.4 COMMAND FLIP-FLOPS

These are the FWD (ref: D/2) and FHALT (ref: F/6) flip-flops.

FWD This flip-flop remembers that a CIO start command has been accepted.

Set: by ACIO and in this condition will activate the CSO that sends the FWDPOC command to the device. When used to drive a card reader, this command is delayed by 10 milliseconds if a card has just been read.

Reset: by either FHALT, ACP or MCL. For the tape reader the reset condition activates the CSO that sends the STOP command to the device. This command is not used by the card reader.

FHALT This flip-flop remembers that either an HIO command has been accepted or that an error condition has been detected.

Set by either FHALTZ1 or ACHALTA. In this condition it resets FWD and inhibits any exchange request.

Reset by ACSST or MCL.

3.5 CONTROL FLIP-FLOPS

These are the FREAD (ref: D/3), the FTHR (ref: F/5) and FIL (ref: E/6) flip-flops.

FREAD This flip-flop remembers that the strobe signal STR has been received.

Set by STRB. In this condition it allows the CU to switch from the EXT to the ECH state.

Reset by either ACINR or CLEAR

FTHR This flip-flop remembers that a throughput error has been detected.

Set by STRA if the CU is in the ECH state.

Reset by either ACIO or MCL

FIL This flip-flop is used to indicate that the number of data characters is wrong when the CU is used with a card reader.

Set by FILZ1A if FHALT is not set when ACP is low.

Reset by either ACIO or MCL

3.6 SEQUENCE TIMING

Figure 3.2 shows the sequence timing of the CU during data transfers. The main differences between tape and card reader exchanges are in the timing of the STR pulse and in the resetting of the signal FWD.

TAPE READER SEQUENCE It can be seen from figure 3.2 that FWD is set by the CIO command and remains set until it is reset by FHALT. This means that once the tape has started to move forward, it will continue to move forward until FHALT causes it to stop; due to either an HIO command being accepted or an error condition being detected. Therefore the CPU must be programmed to accept the data characters in the buffer register before the next strobe pulse arrives at the CU, otherwise throughput errors will occur.

CARD READER SEQUENCE When the CU is used to drive a card reader, FWD is reset by ACP going high. The time between successive strobe pulses is even shorter, so again careful programming is needed to avoid throughput or incorrect length errors.

3.7 DEVICE INPUT SIGNALS

All these signals are active high and are interfaced with the CU logic via special circuits CSI (ref: B/6-11).

CP is the card present signal (ref: A/6) and it will remain high as long as the card covers the read station.

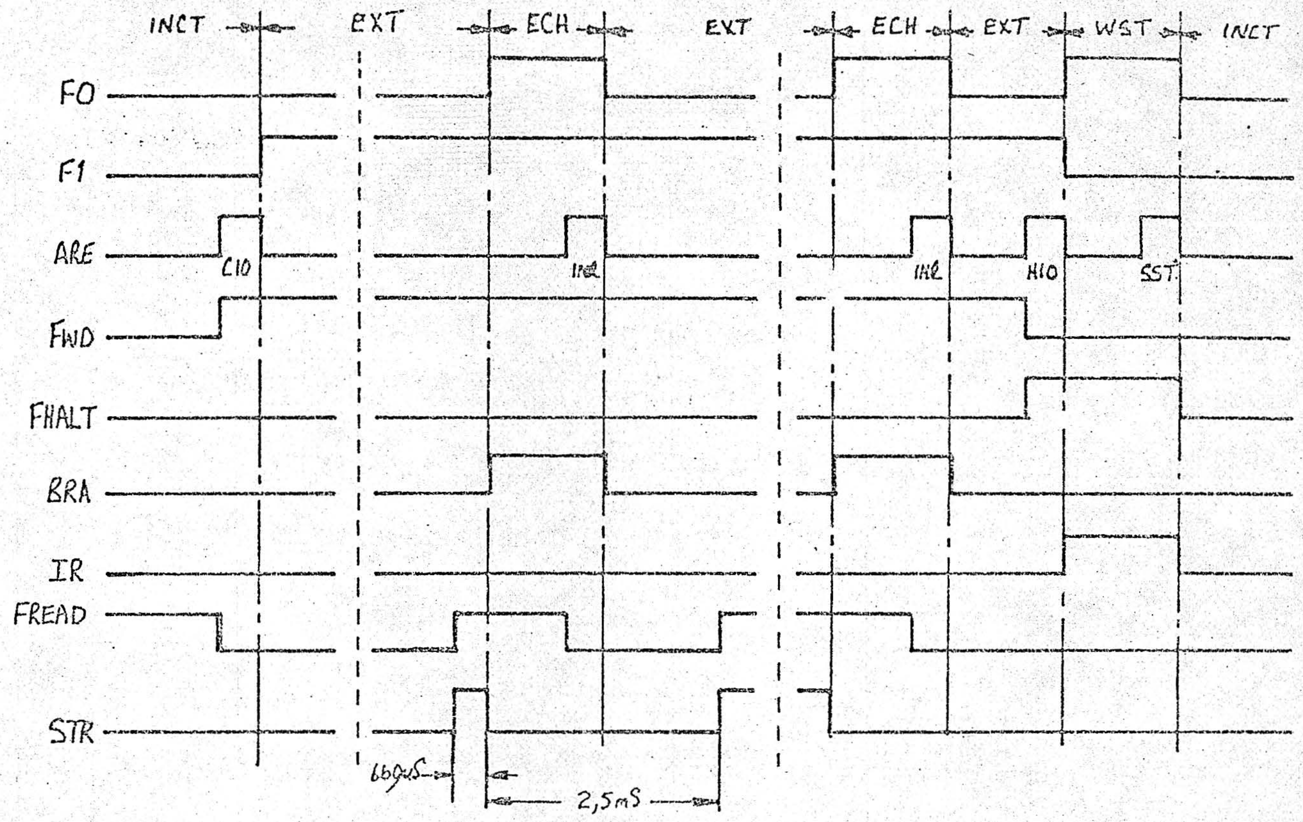
OPER is the card reader operable signal (ref: A/6).

LTLOPER is the load tape lever signal (ref: A/7).

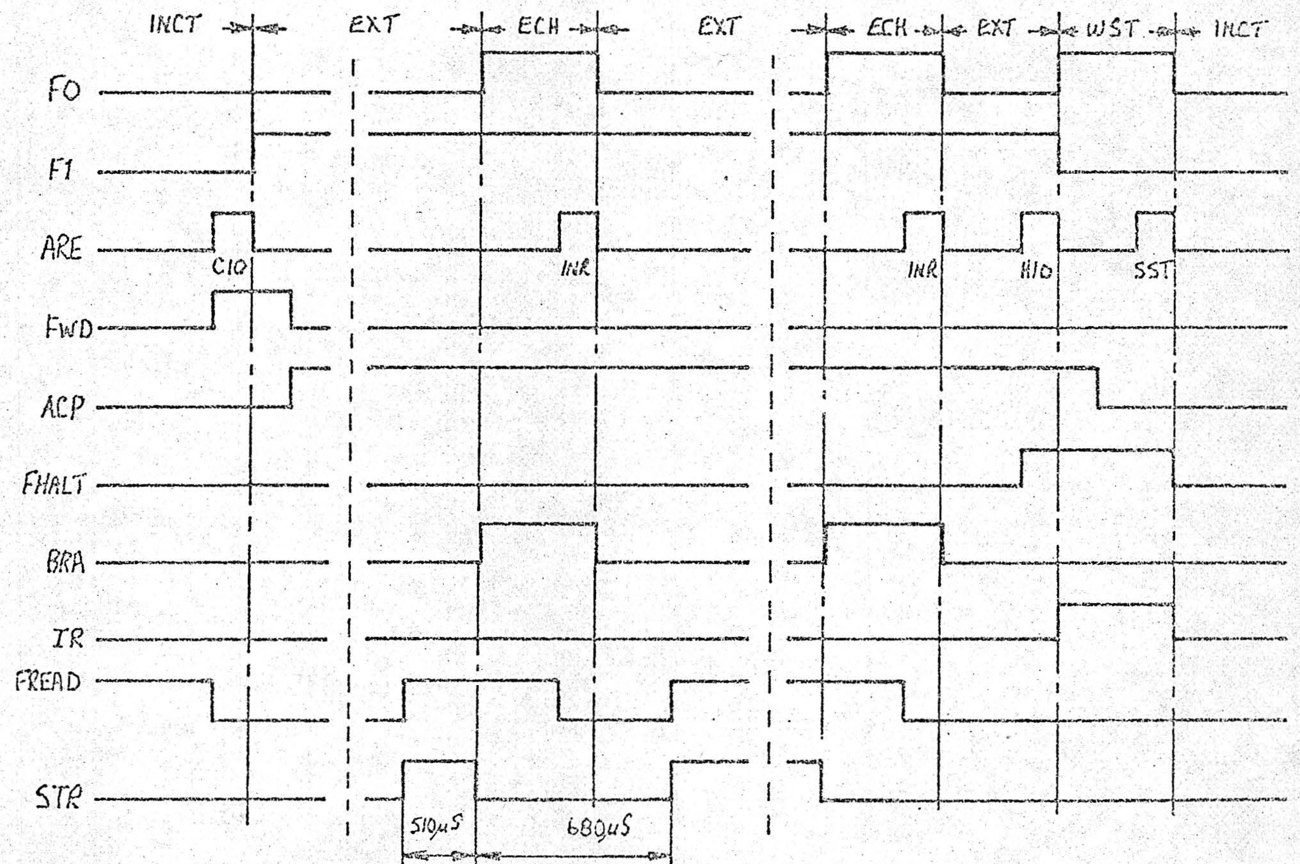
EOT is the end of tape signal (ref: A/7)

PKTBZS is either the SPKTB signal from the tape reader or the ZS signal from the card reader.

Timing for Tape Reader Exchanges



Timing for Card Reader Exchanges



Note: The section between the continuous dotted lines will be repeated as necessary until the end of the transfer.

Figure 3.2 Control Timing

DATA CHANNELS The remaining CSI signals are the data channels, eight from the tape reader and twelve from the card reader. These signals are loaded into the buffer registers.

3.8 BUFFER REGISTERS

These registers G4, F4 and H4 (ref: C-D/8-11) hold the data signals until they are transferred to the CPU by the INR command. The data is clocked into the register by strobe pulses STRA and STRB for the tape reader and by the STRA, STRB and STRC for the card reader. The outputs from the registers are input to one side of the BIN gates.

3.9 BIN GATES

The inputs to BIN gates 15, 14, 12 and 10 come from the buffer registers via logic that also provides status signals. All other BIN inputs come direct from the buffer. Data at these gates is enabled on to the BIN lines by ACINR for the first type of input and by ACINRA for the second type of input.

3.10 STATUS GATES

These gates are F2, F3 and E3 (ref: E/8-9) allow both the CU and the device status to be gated on to the BIN lines in response to either an TST or SST command from the CPU. The BIN lines used to indicate the status are:

BIN 15 a high on this line indicates that either the CU is busy or the device is not operable.

BIN 14 a high on this line indicates that a throughput error has occcured

BIN 12 a high on this line indicates that an incorrect length error has been detected

BIN 10 a high on this line indicates that end of tape has been detected.

All these status bits can be gated on to the BIN lines by ACSST, additionally BIN 15 can be gated by ACTST.

3.11 INTERRUPT SIGNALS

The gates which produce these signals are E1 and C2 (ref: E-F/2). When the CU is controlled by the programmed channel, both signals are 'ored' together on the I/O bus. The signals are:

PIL/ This signal is sent to the CPU when the CU enters the WST state and cancelled by the SST command.

BRL/ This signal is sent to the CPU when the CU enters the ECH state and is cancelled by the INR command.

3.12 OTHER SIGNALS

These are MC (ref: B/6) and EOR (ref: F/7)

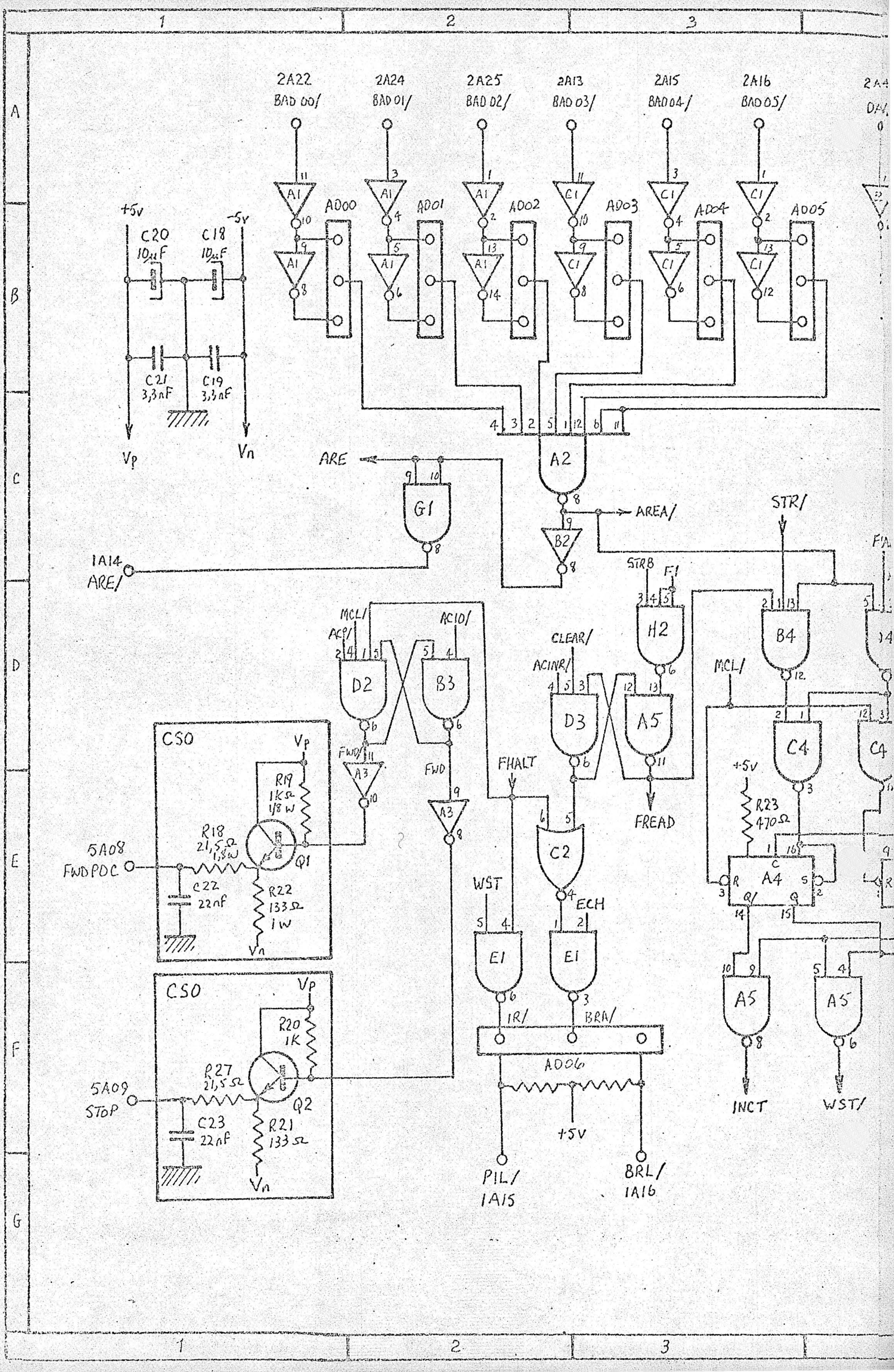
MC This signal is sent from the CPU when either the CPU is switched ON or the operator pushes the MCL button on the control panel. It is used to reset the sequencer, command and control flip-flops of the CU.

EOR This signal is sent by the CPU during multiplex exchanges to indicate the end of the data exchange; it sets the FHALT flip-flop.

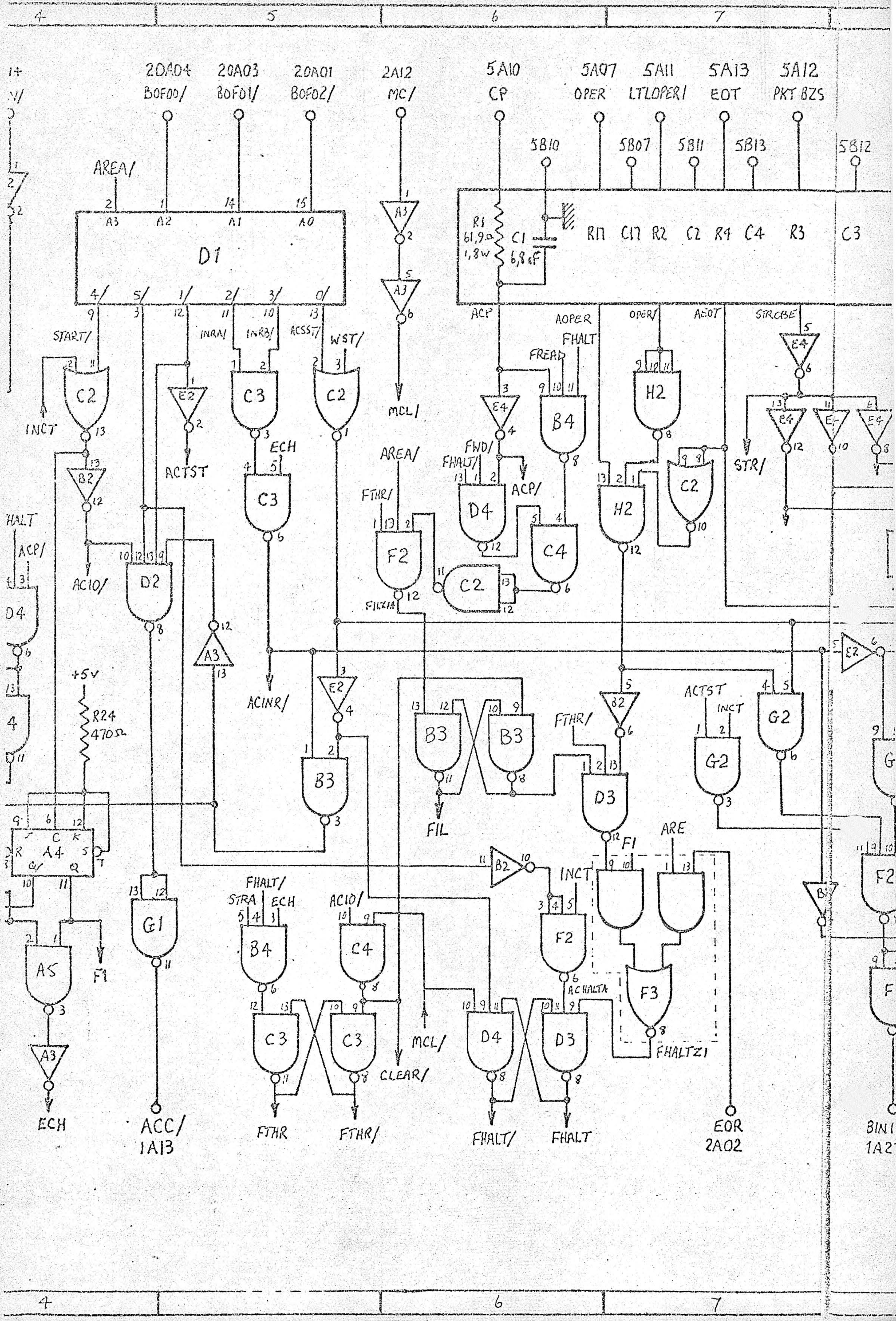
3.13 CU TO DEVICE CONNECTIONS

All connections are made via twisted pairs of wires. One of the wires is connected to ground - both at the CU and the device - and the other to the appropriate signal output. One end of the wires terminates at the Elco connector socket that plugs on to the 5A/B end of the CU card, the other end is connected to the appropriate device socket or card. The following table gives the connections for both the tape reader and card reader. In each case, the lefthand pin is the signal, the righthand pin is ground.

SIGNAL	CU SOCKET	TAPE READER SOCKET J1	CARD READER KAST/800 card
CH1 CH09	5A20/5B20	D/4	n/r
CH2 CH08	5A21/5B21	E/5	h/k
CH3 CH07	5A18/5B18	F/6	c/e
CH4 CH06	5A19/5B19	H/7	Y/a
CH5 CH05	5A16/5B16	J/8	U/W
CH6 CH04	5A17/5B17	K/9	P/S
CH7 CH03	5A14/5B14	L/10	K/M
CH8 CH02	5A15/5B15	M/11	E/H
CH01	5A24/5B24		A/C
CH00	5A25/5B25		t/v
CH11	5A22/5B22		x/z
CH12	5A23/5B23		BB/DD
EOT	5A13/5B13	aa/12	
PKTB ZS	5A12/5B12	N/12	F/J
LTLOPER	5A11/5B11	cc/25	
OPER	5A07/5B07		L/N
CP	5A10/5B10		B/D
FWD POC	5A08/5B08	18/12	R/T
STOP	5A09/5B09	20/12	



2A22 BAD 00/
2A24 BAD 01/
2A25 BAD 02/
2A13 BAD 03/
2A15 BAD 04/
2A16 BAD 05/
AD00
AD01
AD02
AD03
AD04
AD05
+5V
-5V
C20 10µF
C18 10µF
C21 3,3nF
C19 3,3nF
Vp
Vn
A2
AREA/
ARE
G1
B2
1A14 ARE/
STR/
STRB
F1
H2
B4
MCL/
ACP/
AC10/
D2
B3
CLEAR/
ACINR/
D3
A5
FREAD
FWD/
A3
FWD
FHALT
C2
C4
CSO
R19 1KΩ 1/8 W
R18 21,5Ω 1/8W
Q1
5A08 FWDPDC
C22 22nF
R22 133Ω 1W
+5V
R23 470Ω
A4
WST
ECH
E1
IR/
BRA/
AD06
R20 1K
R27 21,5Ω
Q2
5A09 STOP
C23 22nF
R21 133Ω
PIL/ 1A15
BRL/ 1A16
INCT
WST/


4
5
6
7
20A04
BOFOO/
20A03
BOFO1/
20A01
BOFO2/
2A12
MC/
5A10
CP
5A07
OPER
5A11
LTLOPER/
5A13
EOT
5A12
PKT BZS
5B10
5B07
5B11
5B13
5B12
AREA/
A3
A2
A1
A0
D1
R1
61,9Ω
1,8W
C1
6,8µF
R11
C17
R2
C2
R4
C4
R3
C3
START/
INRA/
INRB/
ACSST/
WST/
ACP
AOPER
FHALT
FREAD
OPER/
AEOT
STROBE
INCT
C2
E2
C3
A3
MCL/
ACTST
ECH
E4
B4
H2
FWD/
FHALT/
D4
ACP/
STR/
B2
AC10/
HALT
D2
FTHR/
F2
C4
FILZIA
ACINR/
E2
B3
FIL
FTHR/
D3
ACTST
INCT
G2
+5V
R24
470Ω
A4
A5
F1
G1
ACC/
1A13
ARE
FI
INCT
B2
FHALT/
STRA
ECH
AC10/
ACHALTA
F3
FHALTZ1
MCL/
CLEAR/
ECH
FTHR
FTHR/
FHALT/
FHALT
EOR
2A02
BIN1
1A2

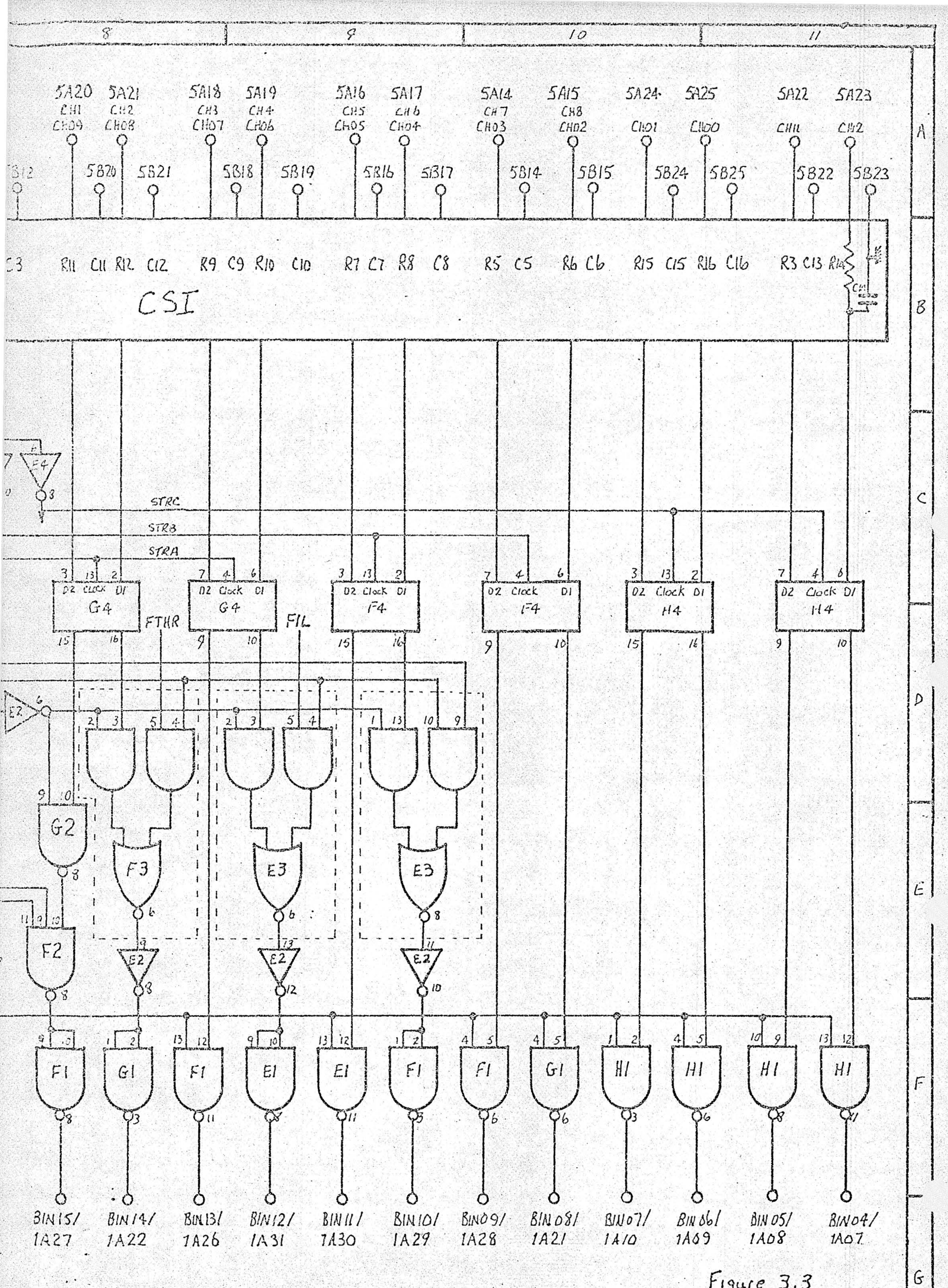


Figure 3.3

Tape/Card Reader Logic Diagram

SECTION IV

TAPE PUNCH CONTROL UNIT

BRIEF DESCRIPTION

A block diagram of the tape punch CU is shown in Figure 4. 1. All the logic and special circuits are contained on one card that is plugged into an I/O bus socket and only one punch can be connected to a card. The special circuits CSI and CSO are used to interface the device to the CU logic circuits. CSI is used for the input signals from the device and CSO is used for the data channels and the command signals to the device. All the special circuit levels are active high. Operation of the CU is controlled by standard I/O instructions and status signals from the device.

The logic and special circuit diagram, which will be found at the end of this section, should be used when reading the following paragraphs. It has grid reference points that are referred to in the text to enable easy location of the circuit elements being described.

4. 1 ADDRESSING (BAD Lines)

The address of the CU can be wired to recognize any desired address using switch connectors AD00 to AD05 (ref: A-B/1-3).

The outputs from these connectors together with the DAV signal from the CPU are gated via element B2 (ref: C/2) to produce the AREA/signal that is used to enable the BOF decoder and the sequencer. It is also used, via element H1 (ref: D/1) to produce the ARE signal that is sent back to the CPU when the address is recognized.

4. 2 FUNCTION DECODING (BOF Lines)

The command from the BOF lines is input to element D1 (ref: A-B/4-5) that decodes the function the CU is asked to perform. If the command is accepted, the ACC signal is sent back to the CPU via H2 and H1 (ref: E-F/4) and the active output level is used to enable other logic elements of the CU. Which element it will be can be seen by refering to the following truth table.

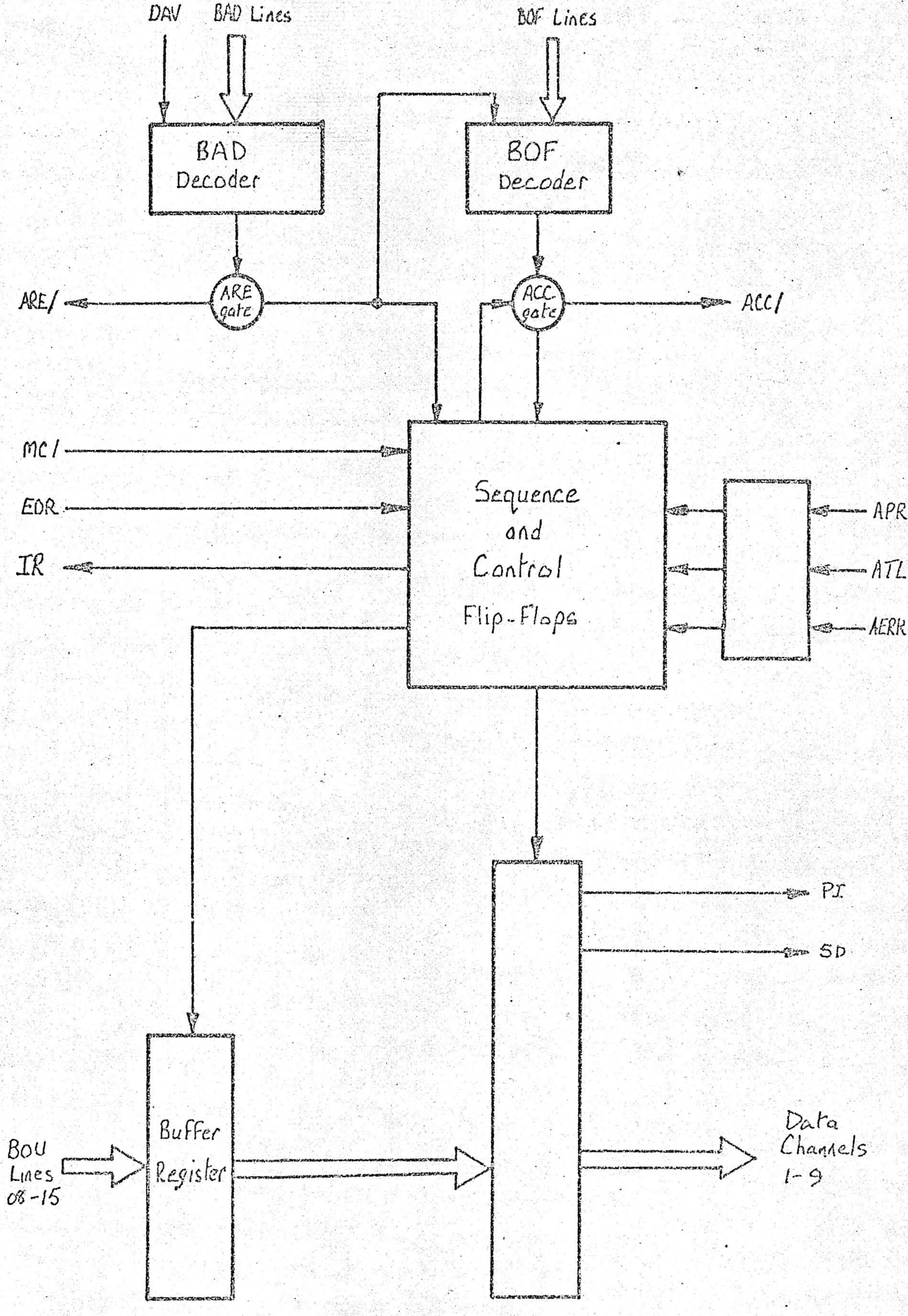


Figure 4.1 Block Diagram of Tape Punch C.U.

A2	A1	A0	Output	(Pin N°.)
0	0	X	OTR	4 + 5
0	1	0	CIO Stop	3
0	1	0	CIO Start	9
1	1	0	TST	12
1	1	1	SST	13

The CIO Stop and the TST commands are always accepted; but before the other commands can be accepted, the CU must be in the appropriate state.

4.3 INTERFACE SEQUENCER

The sequencer uses two flip-flops F0 (ref: E-F/2) and F1 (ref: E-F/3) to switch the CU into one of the four states.

INCT Both F0 and F1 are reset. It is switched into this state either by MC or the trailing edge of an SST command. In this state the CU is waiting for an I/O command.

ECH Both F0 and F1 are set. It is switched into this state by either the trailing edge of the CIO start command or when signal PR goes high at the end of the execute state. In this state the CU askes the CPU for a data exchange and loads the buffer register with data.

EXT F1 is set and F0 has been reset by the trailing edge of the OTR command. In this state the punch instruction is sent to the device and data holes are punched into the tape.

WST F0 is set and F1 is reset by the level FHALT. This will occur when either a CIO stop command is received or an error has been detected. In this state the CU sends an interrupt to the CPU and waits for an SST command. The leading edge of the SST command will reset FHALT and the trailing edge of the command will switch the CU into the INCT state.

4.4 CONTROL FLIP-FLOPS

These are FPI (ref: D-E/3-4) and FHALT (ref: G/6).

FPI This flip-flop remembers that a punch instruction must be performed. One of the outputs from this flip-flop is used via element G3 (ref: E/3) to send the punch instruction to the device.

Set: When the leading edge of the OTR command produces ACOTR via gates E2 and D2 (ref: C/5).

Reset: by either PR, ACSST or MCL

FHALT This flip-flop indicates that an exchange must be ended because either a CIO stop command has been accepted or an error condition has been detected.

Set: by either ACHALT or LTZ1

Reset: by either ACSST or MCL.

4.5 SEQUENCE AND CONTROL TIMING

Figure 4.2 shows the timing of the sequencer and control flip-flops. The time taken to complete one punch cycle may vary slightly from the minimum time shown in the diagram up to a maximum time of 13.3 milliseconds. This variation is dependant up on the time taken by the punch to execute the PI instruction.

4.6 BUFFER REGISTER

This register (ref: B/6-8) comprises two 4-bit latches. Data on BOU lines 08 to 15 is loaded into the register by the clock pulse given by ACOTR. The outputs of the register are connected to the punch data channels via special circuits CSO (ref: C-D/9).

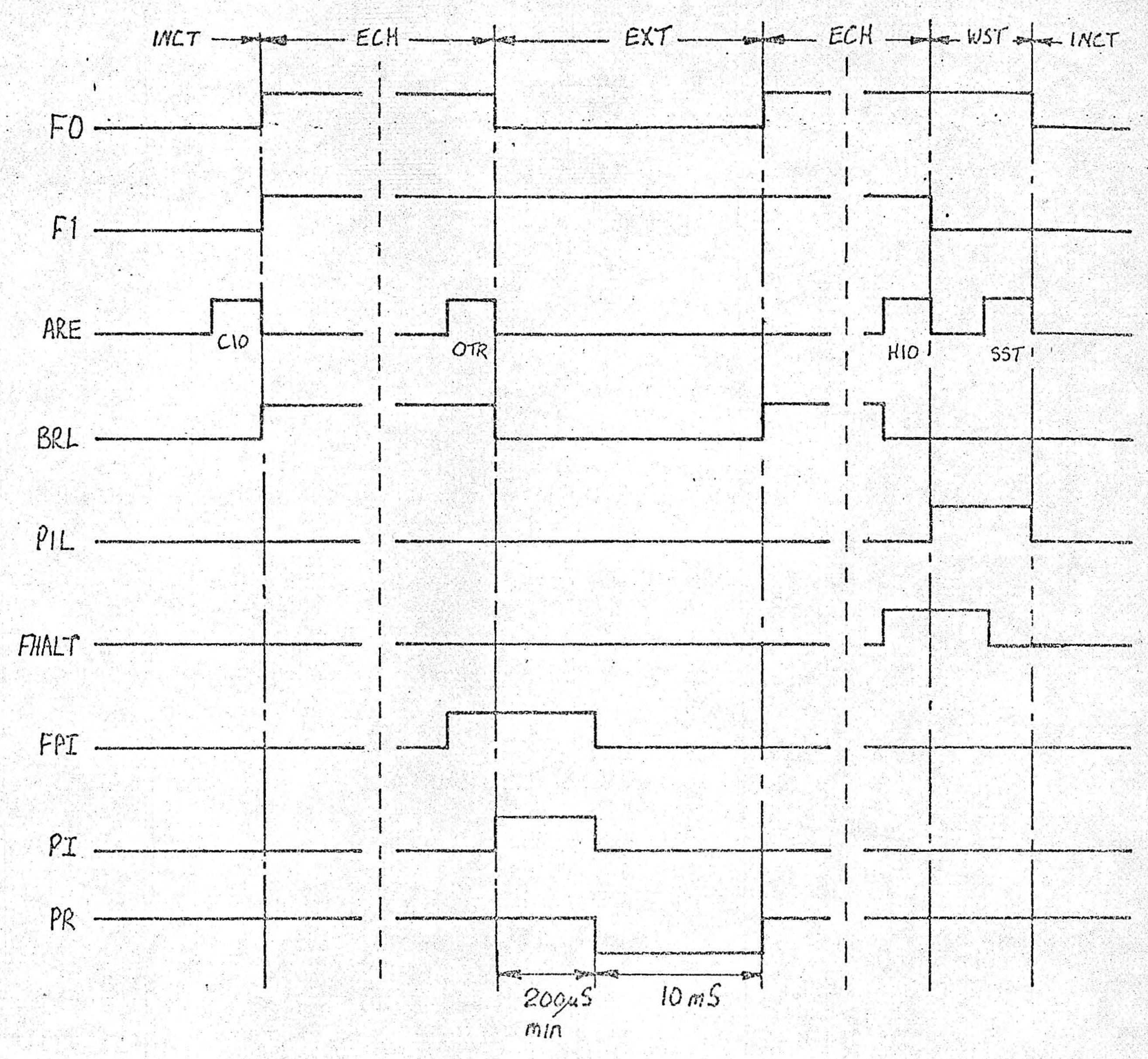
4.7 INTERRUPT SIGNALS

These are BRL (ref: G/4) and PIL (ref: G/3).

BRL This signal is sent to the CPU when the CU switches into the ECH state. It is cancelled by the trailing edge of the OTR command or by FHALT.

PIL This signal is sent to the CPU when the CU switches to the WST state. It is cancelled by the trailing edge of the SST command.

When the CU is controlled by the programmed channel, the BRL and PIL signals are 'ored' together on the I/O bus.



The section between the dotted lines will be repeated as necessary until transfer is ended.

Figure 4.2 Sequence and Control Timing

4.8 PUNCH INPUT SIGNALS

These signals are APR, ATL, and AERR and they are interfaced to the CU logic via special circuits CSI (ref: E/9).

APR Is the punch ready signal. It goes low when data has been stored in the device register and goes high when punching is completed.

ATL Is the tape low signal. It is sent by the punch detects that a new roll of tape is needed.

AERR Is the error signal. It is sent by the punch when either the tape becomes too tight or has broken.

4.9 STATUS GATES

These are gates G1 and E1 (ref: F-G/4-5). The outputs from the gates are sent to the CPU in response to either an SST or a TST command.

BIN15 A high is output to the CPU on this line if either the CU is busy or the punch is inoperable due to an error condition.

BIN10 A high is output to the CPU if the tape low signal has been sent from the punch.

4.10 OTHER SIGNALS

These are SD (ref: D/9), EOR (ref: G/6) and MC (ref: D/1).

SD Is the stepping direction signal to the punch. It should always be low to indicate forward.

EOR This signal is sent by the CPU during multiplex exchanges to indicate the end of a data exchange; it sets the FHALT flip-flop.

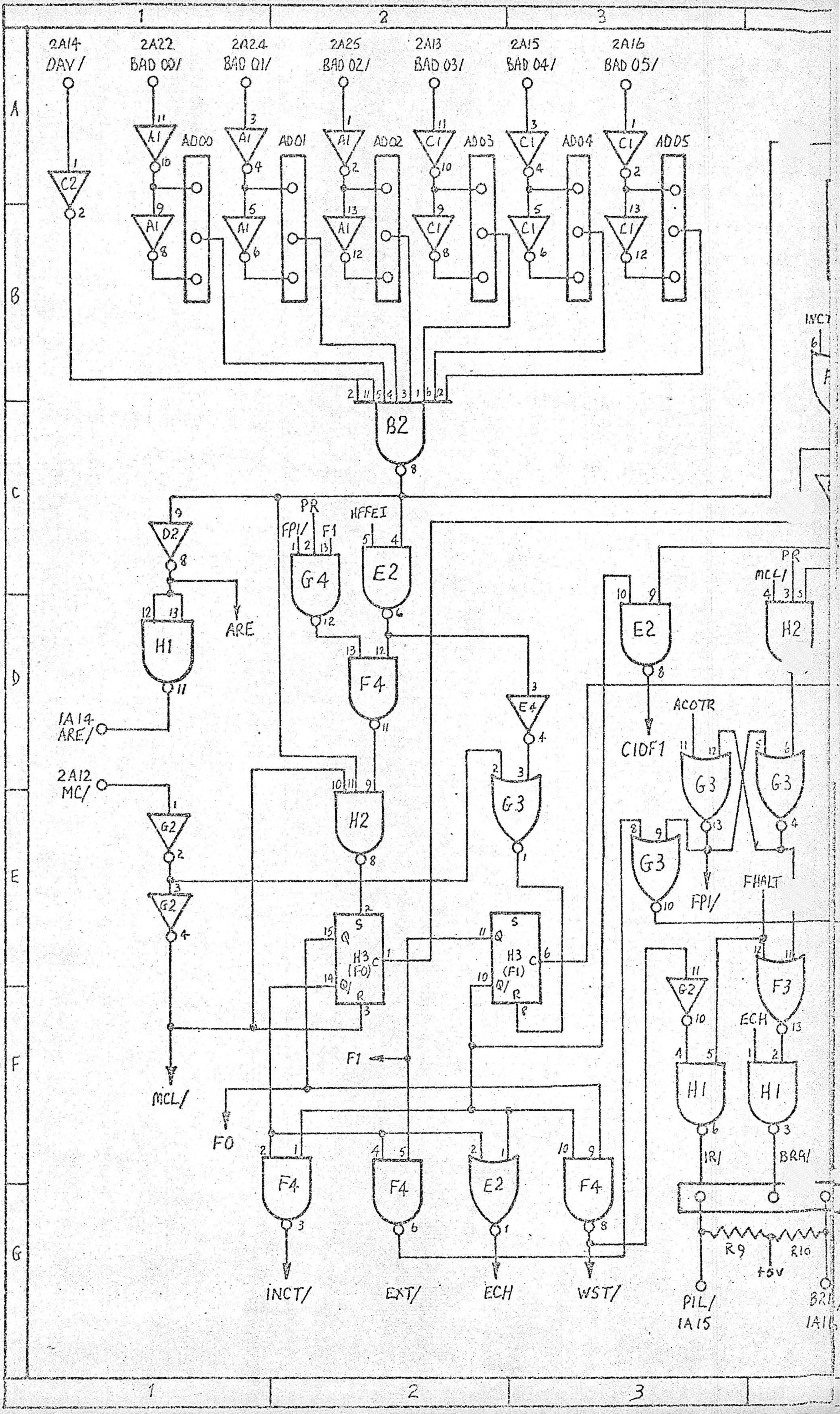
MC This signal is sent from the CPU when either the CPU is switched ON or the operator pushes the MCL button on the control panel. It is used to reset the sequencer and control flip-flops.

4.11 CU TO PUNCH CONNECTIONS

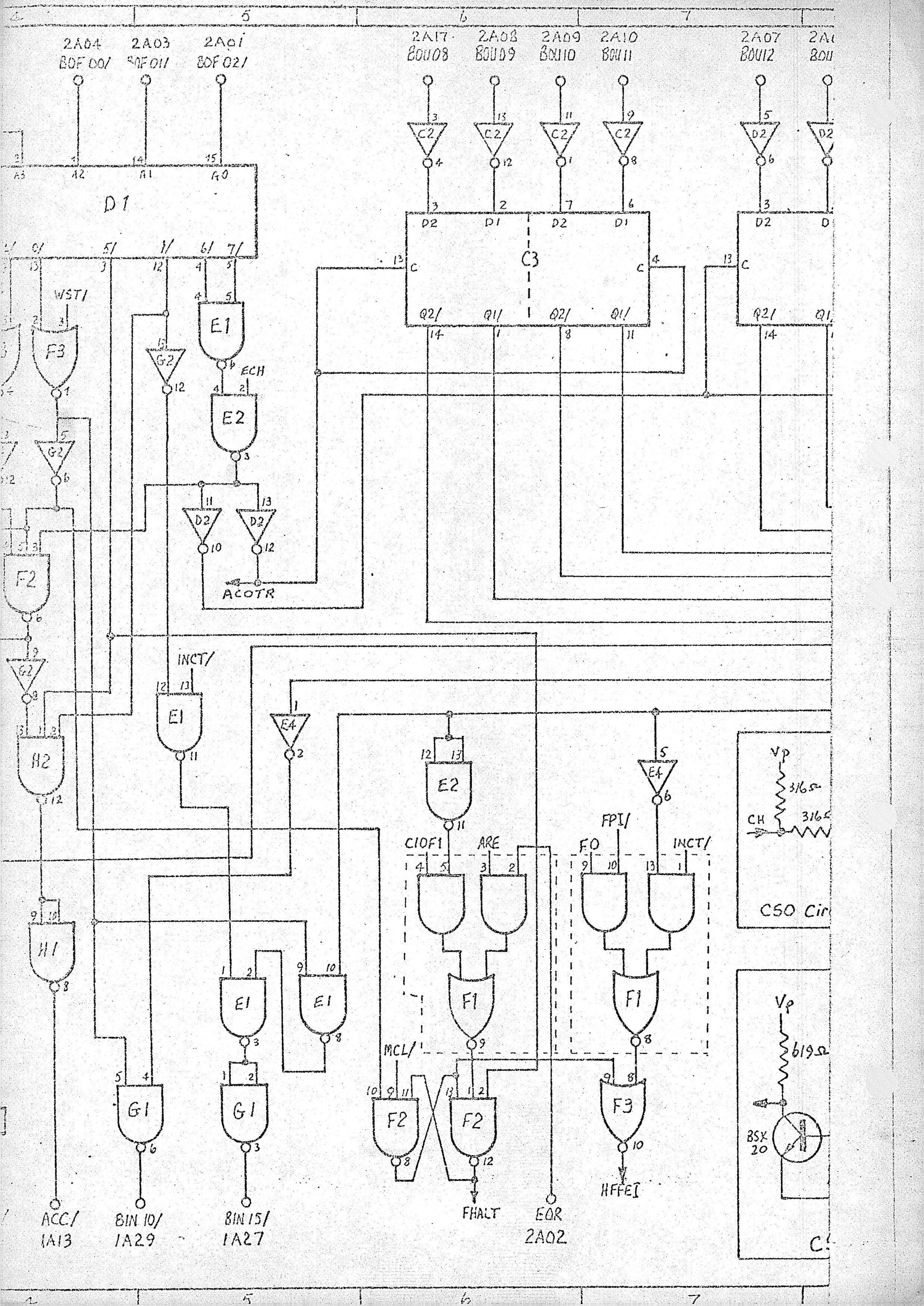
All connections are via twisted pairs of wires. One of the wires is connected to ground - both at the CU and the punch - and the other wire is connected to the appropriate signal output. One end of the wires terminates at the Elco connector socket that plugs on to the 5 A/B end of the CU card, the other end is connected to plug P1 of the punch.

In the following table of connections the lefthand pin is the signal, the righthand pin is ground.

SIGNAL	CU SOCKET	PUNCH PLUG
ACH1	5A01/5B01	1/25
ACH2	5A05/5B05	2/25
ACH3	5A06/5B06	3/25
ACH4	5A04/5B04	4/25
ACH5	5A07/5B07	5/25
ACH6	5A09/5B09	6/25
ACH7	5A10/5B10	7/25
ACH8	5A08/5B08	8/25
ACH9	5A03/5B03	9/25
API	5A02/5B02	11/25
SD	5A11/5B11	10/25
APR	5A21/5B21	12/25
ATL	5A12/5B12	21/25
AERR	5A13/5B13	20/25



1
2
3
A
B
C
D
E
F
G
2A14
DAV/
2A22
BAD 00/
2A24
BAD 01/
2A25
BAD 02/
2A13
BAD 03/
2A15
BAD 04/
2A16
BAD 05/
C2
A1
C1
AD00
AD01
AD02
AD03
AD04
AD05
B2
D2
ARE
H1
1A14
ARE/
PR
FP1/
F1
HFFEI
G4
E2
F4
E4
H2
G3
2A12
MC/
G2
H3
(F0)
(F1)
MCL/
F0
INCT/
EXT/
ECH
WST/
C10F1
ACOTR
FP1/
FHALT
F3
IR/
BRA/
R9
R10
+5V
PIL/
1A15


2A04
BOF 00/
2A03
BOF 01/
2A01
BOF 02/
D1
2A17
BOU08
2A08
BOU09
2A09
BOU10
2A10
BOU11
2A07
BOU12
C2
D2
C3
WST/
F3
E1
ECH
E2
G2
F2
ACOTR
INCT/
H2
E4
C1OF1
ARE
FO
FPI/
F1
H1
G1
MCL/
FHALT
EOR
2A02
HFFEI
ACC/
1A13
BIN 10/
1A29
BIN 15/
1A27
Vp
316Ω
CH
CSO Cir
619Ω
BSX 20

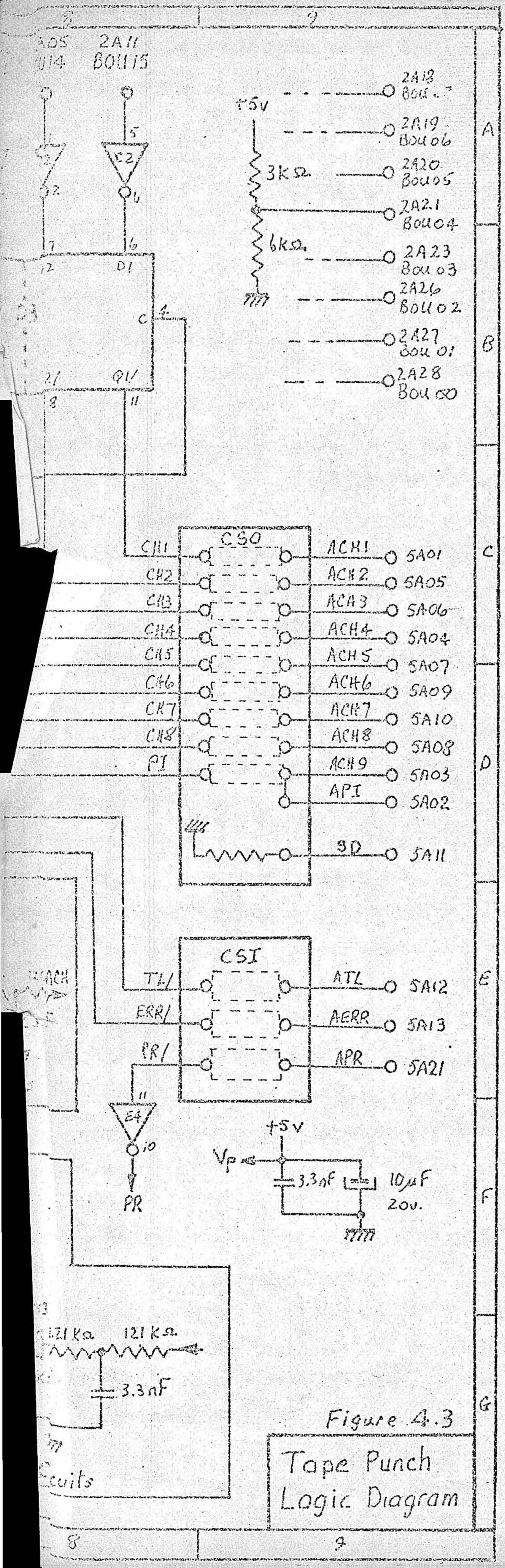


Figure 4.3

Tape Punch Logic Diagram